



SEARCH THEORY APPLIED TO
PARAMETER SCAN OPTIMIZATION PROBLEMS

by

DEAN CHARLES KARNOPP

S. B. and M. S. Massachusetts Institute of Technology

(1957)

Submitted in Partial Fulfillment

of the Requirements for the

Degree of Doctor of

Philosophy

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

June, 1961

Signature of Author
Department of Mechanical Engineering, May 18, 1961

Certified by
Thesis Supervisor

Accepted by
Chairman, Departmental Committee
on Graduate Students



Room 14-0551
77 Massachusetts Avenue
Cambridge, MA 02139
Ph: 617.253.2800
Email: docs@mit.edu
<http://libraries.mit.edu/docs>

DISCLAIMER

MISSING PAGE(S)

Page 122 is missing from the original copy of this thesis. It may just be a blank page and not missing any content.

ABSTRACT

SEARCH THEORY APPLIED TO
PARAMETER SCAN OPTIMIZATION PROBLEMS

DEAN CHARLES KARNOPP

Submitted to the Department of Mechanical Engineering on May 13, 1961, in partial fulfillment of the requirements for the degree of Doctor of Philosophy.

In the first chapter the concept of an optimization problem is introduced by means of a series of simple examples. It is demonstrated that the logical structure of these problems is the same as that of the problem of finding the absolute maximum or minimum of a function defined on a bounded space of many dimensions. The examples show that the parameter space of optimization problems may be discrete, continuous, or mixed and that the functions may be multiply peaked, discontinuous, and time-varying.

In the second chapter an appropriate search theory is studied in some detail. The historical deterministic search methods and their limitations and the decision theoretic aspects of the problem are discussed. In particular, the use of simple but powerful pattern recognition schemes to provide information about the function in the large is suggested.

In the third chapter various concepts of optimum and self-optimizing systems are discussed as well as self-organizing and learning systems. A fundamental limitation to strategies adopted by optimizing controllers analogous to a decision theoretic limitation discussed under the search theory section is presented. The Dynamic Extremal Control is suggested as the prototype of a reasonable solution to the adaptive control problem and is shown to have the effect of automatic behavior pattern recognition.

In the last chapter the role a high-speed analog computer can play in solving a large class of dynamic optimum system problems is demonstrated. The great advantage in memory and logic requirements inherent in simple versions of random searching methods is discussed.

Thesis Supervisor: Henry M. Paynter

Title: Associate Professor of Mechanical Engineering

ACKNOWLEDGEMENT

The author gratefully acknowledges the assistance of his thesis committee in the formulation and completion of the program of study reported here. The committee consisted of Professor Henry M. Paynter, who acted as chairman, and Professors Norman C. Dahl and Thomas B. Sheridan. In particular, Professor Paynter must be credited with the ideas which formed the basis of the initial stages of the investigation, and his enthusiasm was a great help in sustaining the effort until the study acquired a momentum of its own.

The assistance of the Foxboro Company both in supporting part of the research and in constructing some equipment for use with the M.I.T. analog computer is much appreciated.

TABLE OF CONTENTS

ABSTRACT	ii
ACKNOWLEDGEMENT	iii

CHAPTER	PAGE
1 Introduction to Optimization Problems	1
1/1 Concepts	1
1/1/1 Comparability	3
1/1/2 The Space of Alternatives	5
1/2 Examples	6
1/2/1 Static Problems	7
1/2/2 Static-Dynamic Problems	8
1/2/3 Dynamic Problems	9
1/3 General Characteristics	10
2 Search Theory	12
2/1 Introduction	12
2/2 Static Functions	13
2/2/1 The Search Space	13
2/2/2 Location and Value	17
2/2/3 The Role of Trial and Error	18
in Problem Solving	
2/2/4 Decision Theoretic Aspects of Search	24
Problem Formulation	24
Strategy Problems	29
The Role of Experiment	37
2/2/5 Highly Deterministic Methods	40
2/2/6 General Random Methods	44
A Hierarchy of Random Scan Schemes	44
Value Convergence	53
Location Convergence	61
A Comparison Between a Random	63
and a Highly Deterministic Scheme	
2/2/7 Pattern Recognition Schemes	69
2/3 Time Variable Functions	75
2/3/1 Complexity, Strategy and Achievement	77
2/3/2 Modification of the Random Techniques	80
for Time Variable Functions	
3 Optimum and Adaptive Systems	88
3/1 Suggested Optimum Systems	88
3/2 Adaptive Systems	92
3/2/1 Adaptation versus Design	94

3/2/2	Examples of Adaptive Systems	98
	Biological Systems	98
	Man-Made Adaptive Systems	103
3/2/3	Dynamic Extremal Control.	109
4	The Analog Computer in Optimization Studies	119
4/1	Suitable Problems	119
4/1/1	Accuracy	120
4/1/2	Speed	123
4/1/3	Cost	126
4/2	Logic and Memory	128
4/2/1	Advantages of Simple Random Methods	129
4/2/2	Techniques	130
	Timing and Gating	130
	Memory	134
	Machine Output and Display	136
	Appendix: Two Strategies for Long Division	139
	Summary of Experimental Techniques and Results.	142
	Bibliography	149
	Biographical Note	

LIST OF FIGURES

Figure		Page
1-1	An Implication Diagram	7i
1-2	Static Functions	7ii
1-3	Traction Diagram	7iii
1-4	Dynamic Vibration Absorber	8i
1-5	Hydro Electric Governor	8ii
1-6	A Conditional Stability Problem	8iii
1-7	A Dynamic Optimization Problem	9i
2-1	The Effect of Transformation of an Unbounded Space to a Bounded Space	14i
2-2	Some Two Dimensional Search Spaces	15i
2-3	One Way to Search an Arbitrary Bounded Region	15ii
2-4	Location Convergence vs Value Convergence	17i
2-5	A Minimax Search	34i
2-6	Pure Random Scan vs Stratified Random Scan	39i
2-7	Scan Probability Density Functions	45i
2-8	An Example of a Monte Carlo Method.	49i
2-9	A Narrow Range Moving Mean Scan	50i
2-10	A Composite Scan	50i
2-11	An Optimum Narrow Scan.	52i
2-12	The Definition of $P(F)$	54i
2-13	$P_n(F^*)$ For Several n	54ii
2-14	The Expected Value of F^*	55i
2-15	Two Examples of $P(F)$	55i
2-16	$P(F)$ For Parabolic Peaks.	57i
2-17	Definition of $P_1(F^*)$	57i
2-18	Value Convergence with a Narrow Scan	59i
2-19	Comparison of Narrow and Wide Scan Convergence	60i
2-20	Example of a Gross Location Error	63i
2-21	Histograms for Pattern Recognition	73i
2-22	Generation of Correlated Trial Points	73i
2-23	Correlation of F Values	75i
2-24	Dispersion Error	84i
2-25	Lag Error and Dispersion Error.	84i
2-26	Transfer Probabilities	85i
2-27	Composite Scan for Time Variable Problems	85i
3-1	One Type of Adaptive Control	95i
4-1	One Cycle of an Automatic Analog Computer.	130i
4-2	Typical Record of Analog Computer Operation.	130i

4-3	Generation of Composit Scans	132i
4-4	Generation of Binary Parameters	133i
4-5	Discrete Parameters for Organization Studies	133i
4-6	Time Scaling Function Memory	135i
A-1	Standard Division Algorithm	139i
A-2	Minimax Division Algorithm	139i
A-3	Demonstration Apparatus	142
A-4	Two Screen Functions used in Demonstration	143i
	Apparatus	
A-5	Experimentally Obtained Histograms	144
A-6	Static Test Function.	145
A-7	x* Trajectories from Analog Computer Studies	145i
A-8	Block Diagram for Static-Dynamic Problem	147i

On a high hill
Ragged and steep, Truth dwells, and he that will
Reach her, about must, and about must go;

John Donne: Satire III

CHAPTER 1

INTRODUCTION TO OPTIMIZATION PROBLEMS

1/1 CONCEPTS

An optimization problem is involved whenever a choice must be made between alternative courses of action and there exists some way to compare these alternatives. An optimum course of action is one which compares favorably with all other possible courses of action according to the criterion used.

In its broadest form the optimization problem is central to almost all forms of human activity. In fields such as philosophy the concern is primarily with the statement of the goals or criteria for judgment of the alternatives. In fields such as politics or engineering, the concern is more with finding that course of action which will best insure the attainment of accepted goals.

This thesis will concern itself with solution methods for well-stated problems. The preoccupation with mathematical statements of problems is necessary to avoid the ambiguities associated with real-life problems. It is quite possible in technical fields to state problems with precision, but the abstraction involved in doing so carries with it some dangers. Clearly, the real world is too complex to allow optimization problems to be stated in full generality. Not only must the number and kind of alternatives be restricted, but also the criteria for judgment must be simplified and restricted. Another

difficulty arises when the goals of human beings, which may consist of a set of vaguely weighted and contradictory subgoals, must be made precise.

The net result of these considerations is to make obvious the fact that the solution to a formally stated optimization problem may not have a useful relation to the real world. For example, in a computer-controlled war, a formal win for one of the combatants may be disastrous for all concerned. As history shows, human beings are far from infallible in the matter of the solution of optimization problems, but the point is that precise statements of such problems cannot in themselves cure the basic difficulties.

Some of the difficulties associated with real optimization problems can be summarized as follows:

(1) The goals or judgment criteria for an optimization problem are rarely obvious, and hence a precise statement of such goals must be somewhat arbitrary.

(2) The number of alternative courses of action for a real problem is often unimaginably vast.

(3) Since the future cannot be predicted with certainty, all comparisons must involve speculation. The speculation that the solar system will continue on its predicted path seems well justified, but many other necessary speculations are much less well founded.

(4) A peculiarly involuted form of future prediction occurs when

the solution to an optimization problem involves consideration of competitors. Competitors presumably attempt to solve optimization problems with goals contradictory to those of each other.

The statement that an optimum solution has been found should always be received with a certain amount of skepticism until the criterion and alternatives of the problem statement are made clear.

1/1/1 Comparability

Since the heart of an optimization problem is the criterion for comparison, the logic of the comparison process must be well understood. There are many possible ways in which comparisons between alternatives could be made, and some of them could lead to results which were self-inconsistent or at least inconsistent with an intuitive notion of what the solution to an optimization problem should be.

In this thesis the comparisons will be assumed to obey a certain type of logic which is consistent with the properties of numbers, but the criterion functions need not be numerical. The comparisons will be defined first for pair-wise comparisons. In the case of two alternatives, A and B, one and only one of the following three relations is assumed to hold: $A \succ B$, $A = B$, $A \prec B$, where the symbols \succ , $=$, \prec may be given in words such as "is better than", "is indistinguishable from", and "is worse than" respectively. It is also assumed that two alternatives can be compared without any reference to any other possible alternatives. A further assumption is that the relations are

transitive and asymmetric, e. g. , if $A \succ B$ and $B \succ C$, then $A \succ C$.

These assumptions reduce the difficulty of defining an optimum and finding it. Among a finite number of alternatives, A, B, \dots , an optimum, A , is an alternative such that $A \succ X$ or $A \simeq X$ holds for all alternatives taking the place of X . Not all pair-wise comparisons need be tested because of the transitivity condition. If another alternative is found, none of the previous comparisons need be redone, since the relations depend only on the two alternatives tested.

It does not seem that such a logical structure is necessary for comparisons in optimization problems although it is extremely convenient. It would not be surprising to find a human being preferring A to B , B to C , and C to A in a series of pair-wise tests, and yet preferring A to all others when confronted with all alternatives at once. Whether a human's criteria change in time or really do not have the logical structure which is assumed here is not obvious.

There are certain other problems which will arise in the definition of an optimum when an infinite number of alternatives are involved. For instance, if the optimum were defined as the largest number in the set of numbers n , $0 \leq n < 1$, it is clear that no largest number in the set can be given since 1 is excluded from the set. For computational purposes, however, the notions of continuity, infinite precision, and infinite numbers are used merely for convenience so that logical inconsistencies arising in continuum mathematics do not pose a real problem.

For example, the shorthand $A \approx B$ does not mean that A is precisely equal or equivalent to B , but rather that there is not sufficient accuracy to distinguish between them. If such considerations are kept in mind, many inconsistencies associated with infinite sets will be found to have no practical significance.

1/1/2 The Space of Alternatives

Very often it is convenient to make a correspondence between the alternatives in an optimization problem and points in a mathematical space. In this way relations among the alternatives may appear as relations among points in the space and a sort of topology may be discussed.

When there are only a discrete number of alternatives and no particular relations are known to exist between them, the ordering of the alternatives on a discrete space may be artificial and of no great use. However, when the alternatives are expressed by the values of continuous parameters, the space associated with these parameters is almost always a useful mental construct. While the concept of continuous variations of parameters and hence continuous multidimensional parameter spaces is mathematical rather than physical or computational in origin, it is nevertheless useful. This is because the assumption of infinite precision results when finite precision is increased without limit. It often is hard to justify choosing a particular precision for a general study, for the results are often drastically affected by the

inaccuracy or imprecision actually chosen. It often seems more reasonable to state results independent of finite accuracy or precision and then to inquire whether the results are applicable to the real world. Results which require infinite accuracy and precision clearly cannot be justified experimentally in the real world, and engineers are mainly interested in those results which require only the attainable accuracy.

It is also possible to speak, for convenience, of variables which range over an unbounded space, i.e., $-\infty < x < +\infty$. Again this must be a mental construct, since the infinitely large is as impossible to attain physically as the infinitely small. In a practical situation there must be a bound on the parameter space which is actually searched just as there is a bound on the accuracy of computations. When justified, either of these bounds may be shifted by appropriate rescalings of the problem.

1/2 EXAMPLES

Given below are examples of optimization problems. The problems represented are maximization or minimization problems involving functions of several variables. The alternatives of the optimization problems are associated with values of the variables (parameters) and hence with points in the indicated spaces. The comparison between alternatives is based on the value of a function so that the symbols

$>$, $=$, $<$ acquire their usual meaning. The numerical values of the function may or may not have any significance, but the relative values always are important since the comparison is obviously based on relative values.

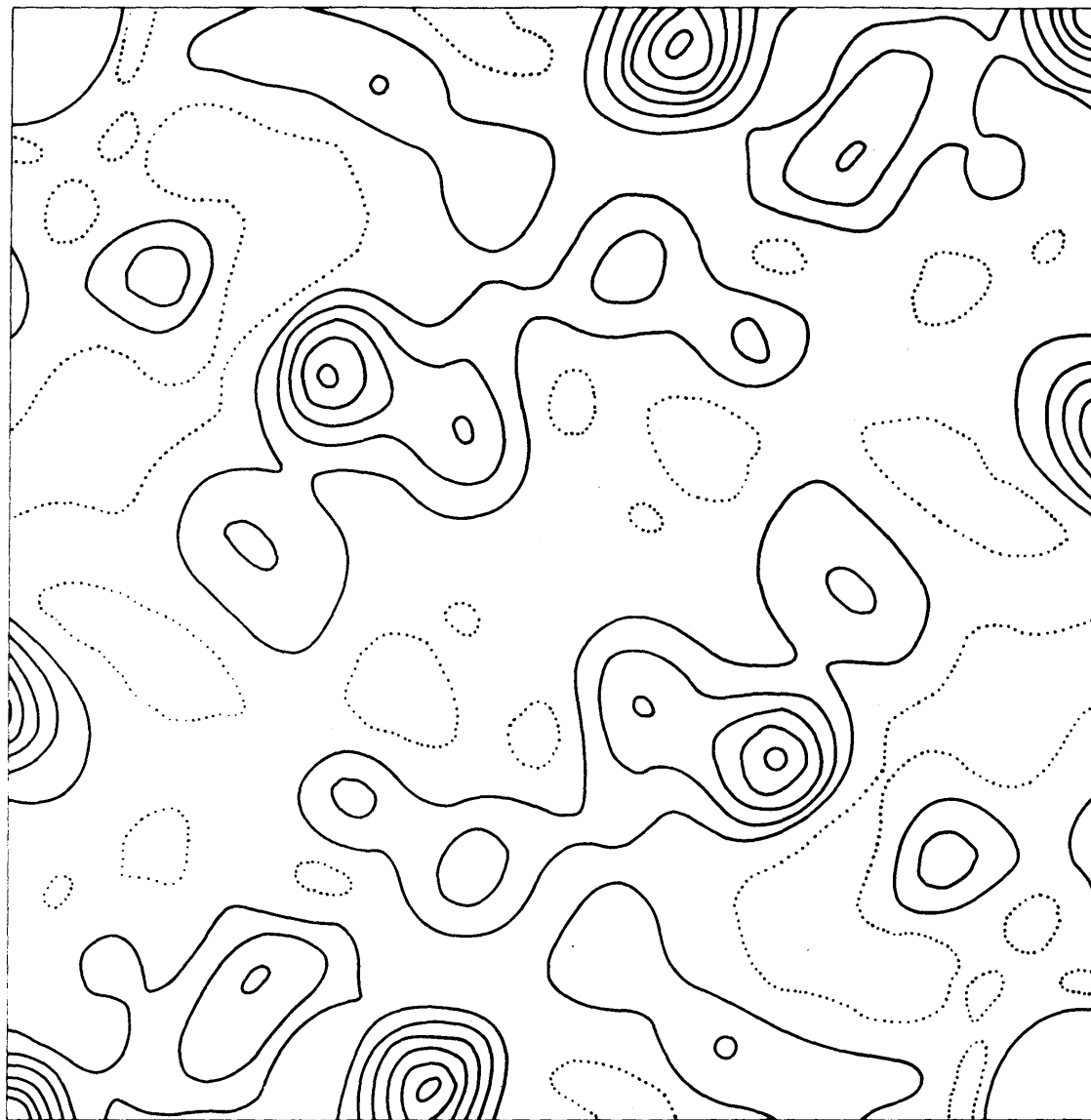
1/2/1 Static Problems

A static problem is a problem in which time is not involved. Neither the comparison between the alternatives nor the space of permissible alternatives should change in time. Any problem which requires the solution of dynamic equations before the comparisons can be made is also excluded.

Figures 1-1, 1-2, and 1-3 show examples of static functions defined on various types of spaces. It would be a static optimization problem of the sort which shall be studied in Chapter 2 to find the absolute maxima of these functions.

0

b/2



a/2

Figure 1-1

An implication diagram for Narsukite, $\text{Na TiO Si}_4\text{O}_{10}$ obtained by transformation of the Patterson function. This type of diagram, which is used in crystallographic studies, can be interpreted as a contour map. The solid lines indicate positive height, and hence maxima, and the dotted lines indicate negative heights and minima. In this complicated function of two variables, the locations of the maxima are of interest and this type of plot is made to aid in the discovery of such locations. This technique cannot be used for a function of more than two variables.

Reference for Figure 1-1: Peacor, D.R., The Crystal Structure of Narsukite, $\text{Na TiO Si}_4\text{O}_{10}$, Masters thesis in geology, M.I.T., June, 1960.

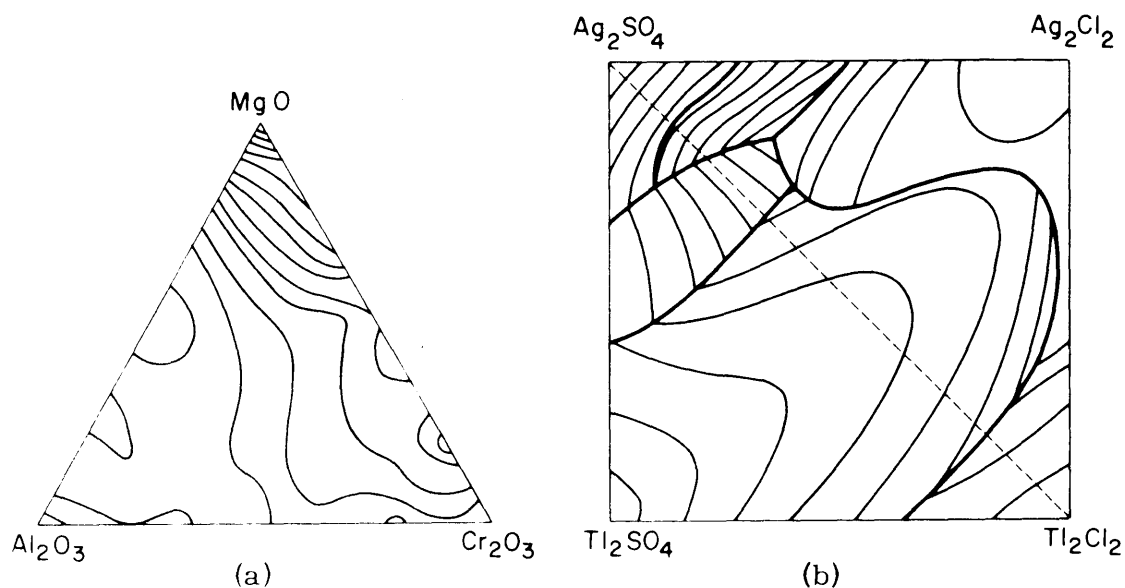


Figure 1-2

Two examples of functions defined on a two dimensional space. Every point in the triangle of Figure 1-2 (a) represents a ternary system composed by particular percentages of each of the three compounds shown at the vertices. In Figure 1-2 (b), two such triangular regions are combined to form a square region. In each case, contour lines are drawn in order to indicate values of a property on the system. In Figure 1-2 (a), there are several maxima and minima, and in Figure 1-2 (b) there are lines of discontinuity in slope.

References for Figure 1-2: Landolt-Bornstein, Zalenwerte und Functionen..., II Band, 3 Teil, (a), p. 221, (b), p.258.

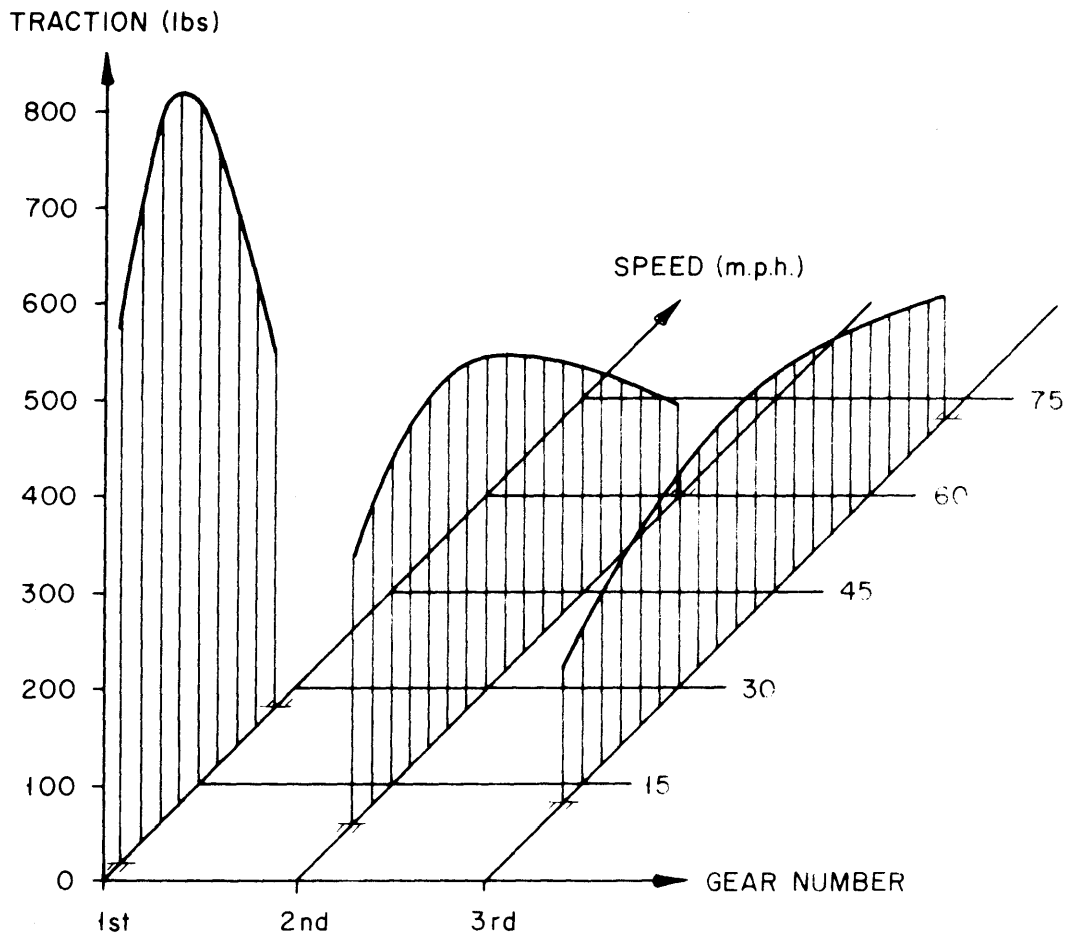


Figure 1-3

Traction Diagram for an Automobile

An example in which one continuous variable, speed, and one discrete variable, gear ratio, are involved. Older search methods would have to treat this as three separate problems, but the methods we propose can treat this as one large problem. The advantage is that if a maximum were being sought, we would not need to pick one subfunction and exhaustively study it only to find later that another subfunction was uniformly higher.

Reference for Figure 1-3: SAAB 93 Service Manual

1/2/2 Static-Dynamic Problems

In static-dynamic problems, dynamic equations (involving time) must be solved before a comparison between alternatives can be made. In other words, the criterion function, while not a function of time, can only be evaluated by operations which explicitly involve time. These problems, which are really static problems, are put into a separate category because they are singularly suited for analog computation, which will be studied in Chapter 4.

Figures 1-4, 1-5, and 1-6 are examples of static-dynamic problems. In these examples the comparison between alternative parameter settings is based on properties of the solutions of dynamic equations. In the simpler cases the properties can be deduced analytically so that computational solution of the dynamic equations is not actually required.

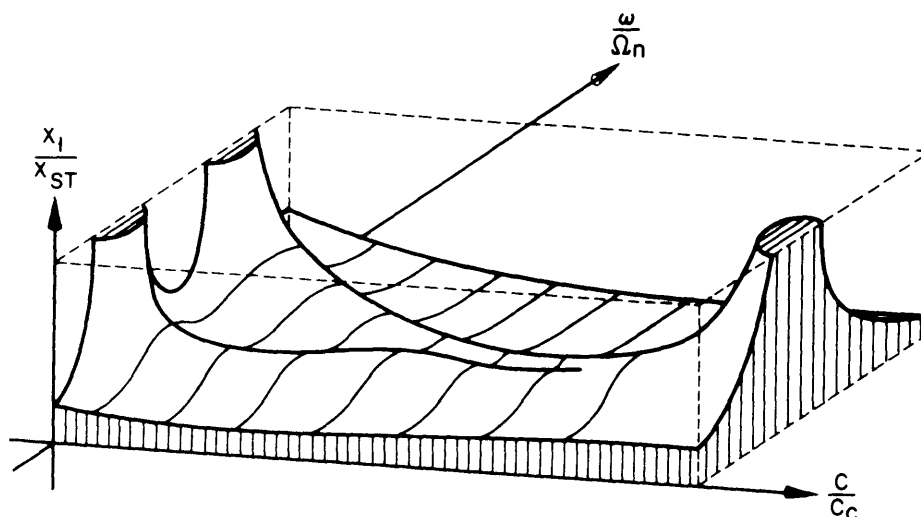


Figure 1-4

Sketch of the effect of a Dynamic Vibration Absorber.

A dynamic vibration absorber is supposed to prevent oscillation of a piece of machinery on a vibratory environment. One way to specify an optimum absorber is to require that the maximum amplitude of vibration found by varying the frequency of the excitation be as small as possible. This minimax type of criterion merely complicates the evaluation of the function to be minimized. In the example of the sketch, we are only able to show one parameter, c , although there are really three.

To evaluate the criterion function, we first pick a value for c and any other parameters to be studied, and then scan the frequency, ω , to find the maximum of x_1 . This maximum of x_1 is the value of the criterion function to be associated with the chosen set of parameters. We then search for that set of parameters which will result in a minimum of the criterion function.

Reference for Figure 1-4: Den Hartog, J. P., Mechanical Vibrations, McGraw Hill, N. Y., 1947, p. 123.

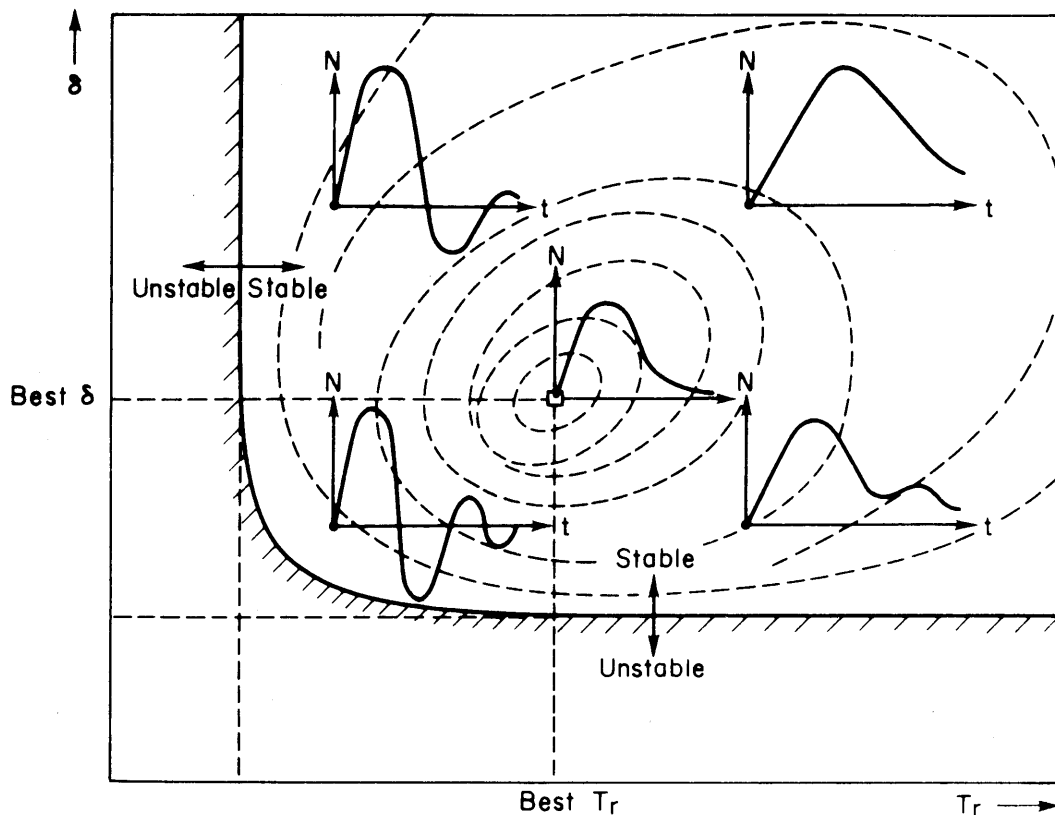


Figure 1-5

Effect of Governor Settings on Speed Recovery of a Hydroelectric Unit

In this sketch, the effect of the variation of two parameters is presented in an unusual way. By inspection, one can see that almost any error criterion such as the integral squared error criterion, will result in a function which is minimum at a point that is very near to the indicated optimum point. The dotted lines indicate contour lines of such a function. In order to evaluate the function, one would first have to solve dynamic equations in order to find the speed error as a function of time. Also one would have to perform such operations as squaring and integrating to carry out the evaluation of the particular error criterion involved.

Reference for Figure 1-5: Paynter, H. M., Methods and Results from M.I.T. Studies in Unsteady Flow, B.S.C.E. Journal, v. 39, n. 2, 1952.

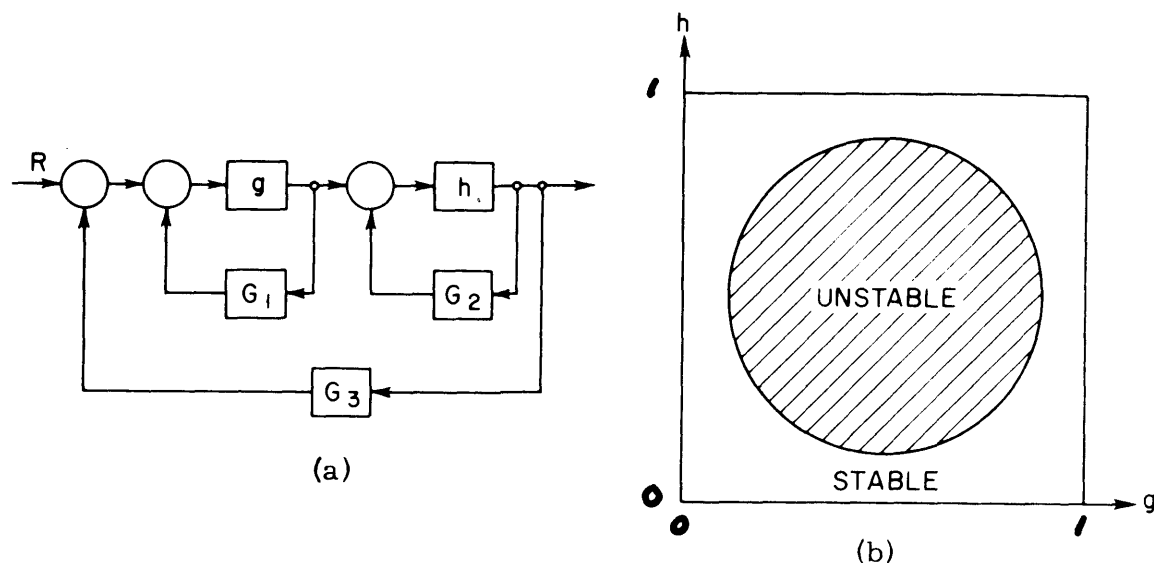


Figure 1-6

A Conditional Stability Problem

In this example, two parameters, the gains g and h , of a linear system are varied and the stability of the system is studied. The interested reader can find from the reference that the operators G_1 , G_2 , G_3 , do not appear to be unusual, and yet the stability region has the peculiar shape of Figure 1-6 (b).

Since the search for an optimum system is more complicated than a search for a stable one, we feel that peculiar results for stability analyses must lead us to expect difficult search problems even when only linear systems are involved. For example, if we had set $h=1/2$ and searched for an optimum value of g , the function must clearly not be monomodal for there are two distinct regions separated by an unstable zone which might contain the optimum. A method suited only for finding relative optima would not be appropriate for such a problem.

Reference for Figure 1-6: Truxal, J. G., "Control Systems-Some Unusual Design Problems", Adaptive Control Systems, E. Mishkin and L. Braun, Jr., eds., McGraw Hill, N. Y., 1961, p. 111.

1/2/3 Dynamic Problems

A dynamic optimization problem is one in which the comparison of alternatives is a function of time. The comparability properties of 1/1/1 are assumed to hold for one instant of time but not over finite intervals. Since comparisons generally must be made sequentially in time, the behavior which accompanies the logical structure of comparison that we have assumed for static problems is largely lost. However, since the comparisons obey the logic of 1/1/1 except for time variation, much can still be done that could not be done if the comparisons obeyed some more complicated logic. There is an analogy here between linear and time-variable linear systems, in which the more complicated behavior of a time-variable linear system still can be treated more easily than a nonlinear system.

A dynamic problem may involve also time-variable bounds on the alternative space, but this is not illustrated.

Figure 1-7 illustrates a dynamic problem which could occur if one of the parameters of a static problem were given as a function of time. Time-variable problems will be discussed at some length in Chapter 3.

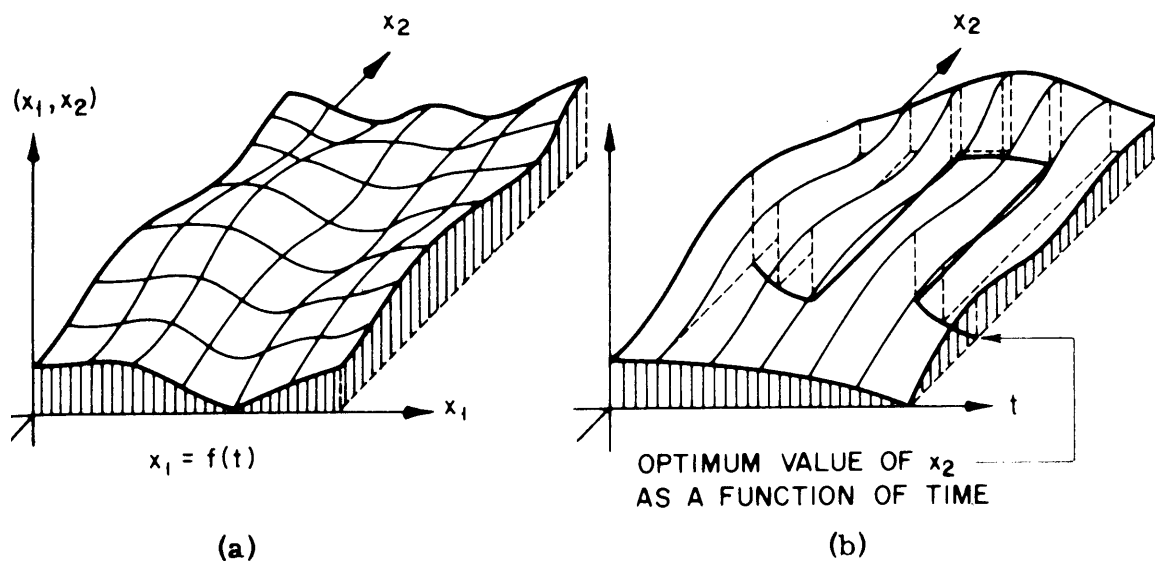


Figure 1-7

A Dynamic Optimization Problem

In Figure 1-7 (a) a function, $F(x_1, x_2)$, is shown which is to be maximized. During the maximization process, x_2 can be varied at will, but x_1 varies spontaneously with time. In Figure 1-7 (b) the resulting time variable search problem is shown. As time goes on, the function to be maximized by searching along the x_2 axis changes.

It is important to note that even if F is continuous, and x_1 changes continuously in time, the value of x_2 which yields the maximum may jump discontinuously. This occurs when a relative peak rises to the level of the absolute peak and exceeds it, thus becoming itself the new absolute peak.

1/3 GENERAL CHARACTERISTICS

The problems which will be discussed have several characteristics. First, the problems have the same logical structure as maximization or minimization problems involving functions of a finite number of variables, possibly including time. Problems involving functionals will have to be reduced by approximation or assumption to problems involving functions. The value of the criterion function need not be numerical but must have the comparability property mentioned in 1/1/1.

Secondly, we assume that any useful technique for reducing the dimensionality of the problem will be used before the problem is stated. For example, Bellman has shown how some maximization problems in m dimensions can be reformulated as m one-dimensional problems. (Bellman¹)* Such a reduction may be very useful and should be used if convenient. However, many problems cannot be reduced at all and must be studied in a space of many dimensions.

A function of several variables will usually be designated as follows: $F(x) = F(x_1, x_2, \dots, x_n)$, that is, we use x to represent the n -dimensional bounded space of the parameters x_1, x_2, \dots, x_n . Other symbols will be defined as they appear. It should be noted that there is no logical

* Names in parentheses thus, (Bellman¹), refer to the bibliography at the end of the thesis. The references are listed alphabetically by the last name of the first author with a superscript if more than one title is listed for an author.

difference in the techniques for searching for an absolute maximum or a minimum, and the words absolute extreme will be used to denote one or the other of these.

CHAPTER 2

SEARCH THEORY

2/1 INTRODUCTION

Everyone who has taken a first course in calculus has been exposed to classical methods for finding the extreme of a function. A typical problem is to fold a sheet of paper into a box containing the largest possible volume. In these problems the function is differentiated to express mathematically the condition that the slope of the function is zero at an extreme. In the simple cases treated the equations obtained can be solved analytically to yield the values of the variables which result in an extreme of the function.

It is clear that these methods and several similar computational methods are severely limited. The limitations will be discussed below. The point to be made here is that analytic techniques obscure the fact that a search is really required to find the extreme of a function. It is also true that even when an analytic solution is available, accuracy considerations may often make a search procedure more attractive for numerical calculations.

On the other hand, there exists a fairly well developed theory of search, and one might expect that this could be applied from the beginning in the problem of a search for an extreme. Much of this theory, developed primarily for the purposes of operations research, is, however, not applicable. For example, in problems such as searching for

a needle in a haystack, a submarine in an ocean, or a white ball among boxes of black balls, the stop rule -- stop looking when you have found what you are looking for -- is trivial. On the other hand, in the search for the maximum of a function or, equivalently, in the search for the top of a mountain range in the dark with only a glow-in-the-dark altimeter, the appropriate stop rule is not evident.

There are many other differences between the problems discussed below and the problems discussed in most well developed search theories. In what follows we shall take the attitude that an extremization problem is a search problem, but our considerations will probably not apply without modification to search problems in other contexts.

2/2 STATIC FUNCTIONS

2/2/1 The Search Space

In this section the problem of searching for the maximum of a function $F(\mathbf{x}) \equiv F(x_1, x_2, \dots, x_n)$ will be studied. The function F is not a function of time, although time may be involved in the operation of evaluating F as in the static-dynamic case of 1/2/2.

The parameters x_1, x_2, \dots, x_n are presumed to be independently variable within some bounded region of the n -dimensional space and may take on discrete or continuous values depending on the problem. All relations of the form $x_i \equiv f_i(x_k, x_1, \dots, x_p)$ should therefore be incorporated in the definition of F so that the minimum number of x_j 's will be involved in the definition of the space \mathbf{x} .

The question of the boundedness of the search space is of considerable importance. If the search space is bounded and if the function is smooth enough, there is a chance that a finite number of trial points may provide useful information about the maximum of the function. However, if the function is imagined to be arbitrarily non-smooth or is defined over an infinite space, the problem may well be theoretically unsolvable in any reasonable sense. In fact, the boundedness of the search space is connected directly with the smoothness of the function. For example, Figure 2-1 demonstrates that a transformation of a smooth function defined on an infinite space to a function defined on a bounded space may not be of any use, since the function may become arbitrarily nonsmooth after transformation.

In any computational or experimental search, the space to be searched must be bounded, but the bounds of the space may be determined in two ways. Sometimes the bounds are fixed rigidly by the problem. In such cases the values of parameters outside of the valid region are meaningless. In other cases the bounds are set for convenience or to represent an a priori belief that the maximum of the function will be found within the region. In these latter cases the bounds may be moved if it appears after some searching that the bounds were poorly chosen.

The simplest type of bounds are those imposed independently on each parameter, e. g., $c_1 \leq x_1 \leq c_2$. The independent bounds generate

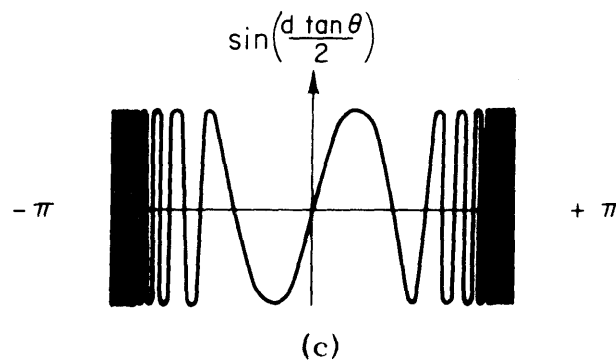
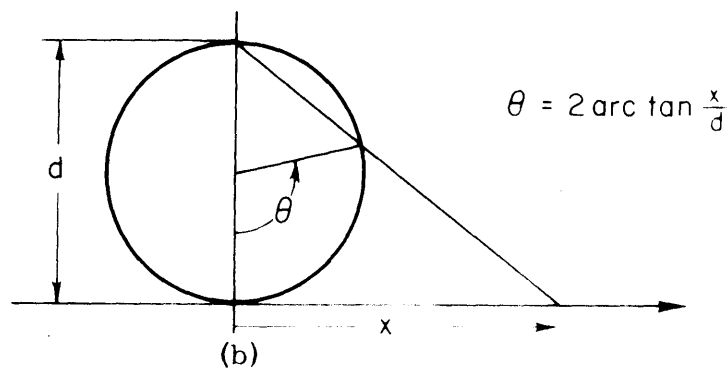
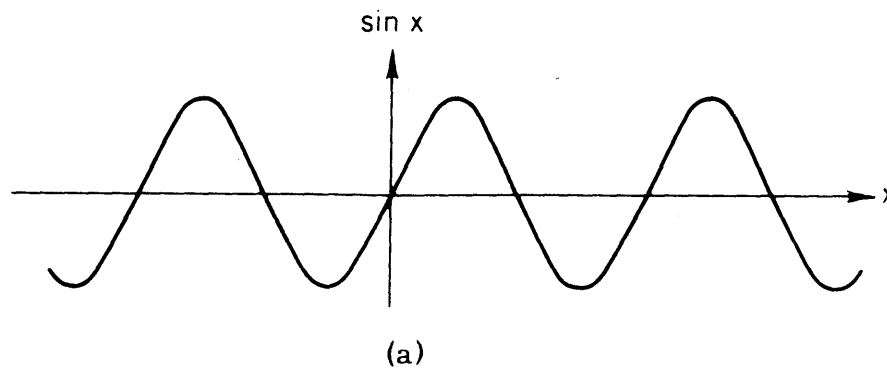


Figure 2-1

An example of a smooth function defined on an infinite space which becomes drastically nonsmooth when the infinite space is transformed into a bounded one. In (a), the function $\sin x$ is defined for $-\infty < x < \infty$. In (b), x is transformed into the bounded variable, θ , and the resulting function of θ is shown in (c).

a hypercubical search space, a two-dimensional version of which is shown in Figure 2-2(a).

More complicated bounds in the search space can be given by functional inequalities involving more than one variable. For example, in Figure 2-2(b) the valid search space is indicated as that region in x_1, x_2 such that $f_1 = ax_1 + bx_2 + c \geq 0$ and $f_2 = d^2 - x_1^2 - x_2^2 \geq 0$. The arrows indicate regions of the space which satisfy the inequalities separately, and the shaded region is the intersection of such regions.

Figure 2-2(c) illustrates the case in which no points in the search space satisfy both bounding relations. Although this situation can arise because of a simple mistake, it can also be the result of a more basic difficulty. It is useless to attempt to search, for example, for an alloy of maximum yield strength under the conditions that it must have less than a certain maximum density and cost, if no alloy satisfies these constraints. In cases such as this there are really two searches involved; one search to find which parameter values satisfy the bounding relations, and another to find which, among the valid parameter sets, results in the extreme of the criterion function.

In some cases these two searches can be accomplished separately. In other cases the searches must be done simultaneously. One possibility is illustrated in Figure 2-3. In this case a large, hypercubical space which contains the valid search space is searched. Whenever a point in

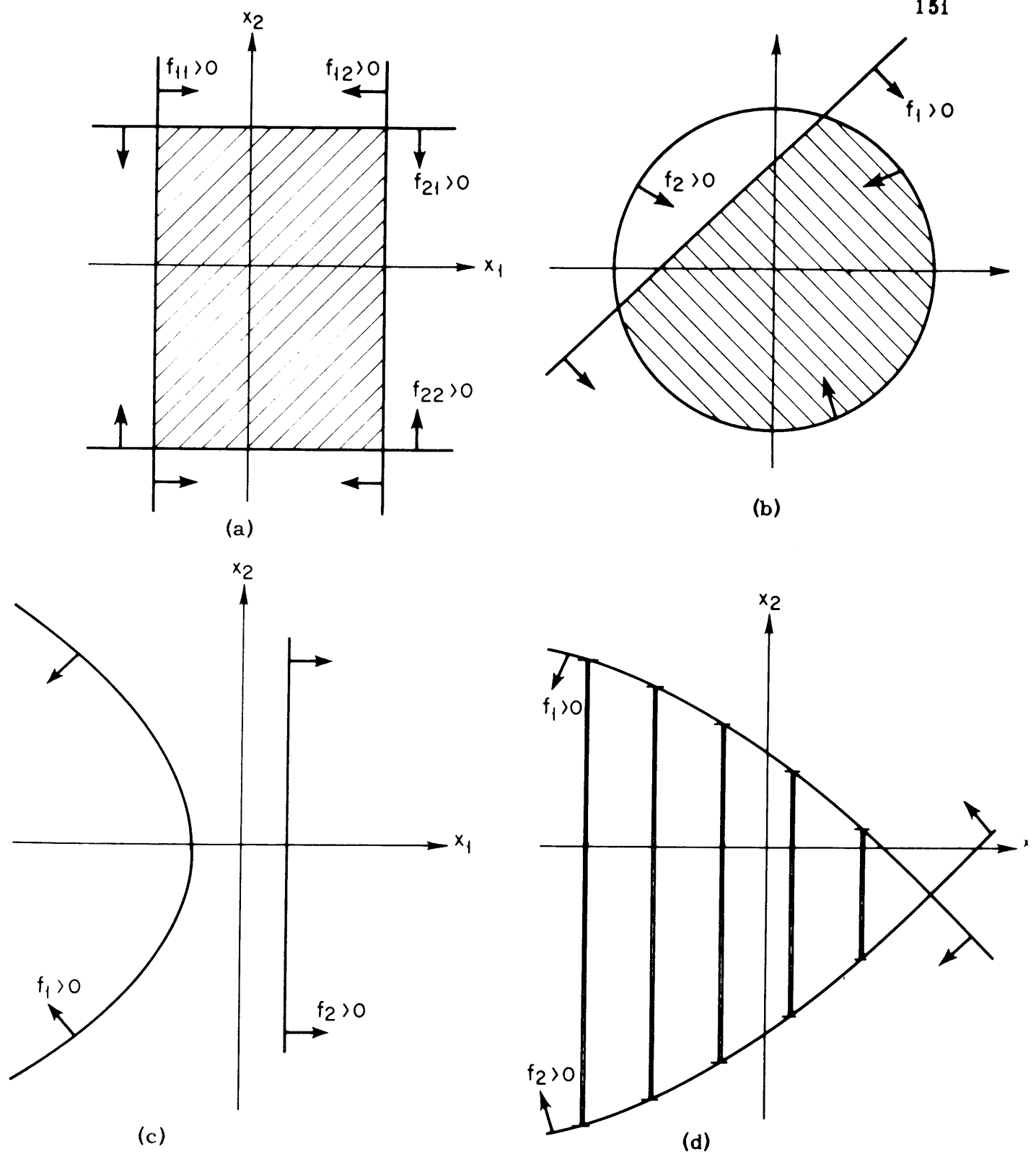


Figure 2-2

Some Two Dimensional Search Spaces

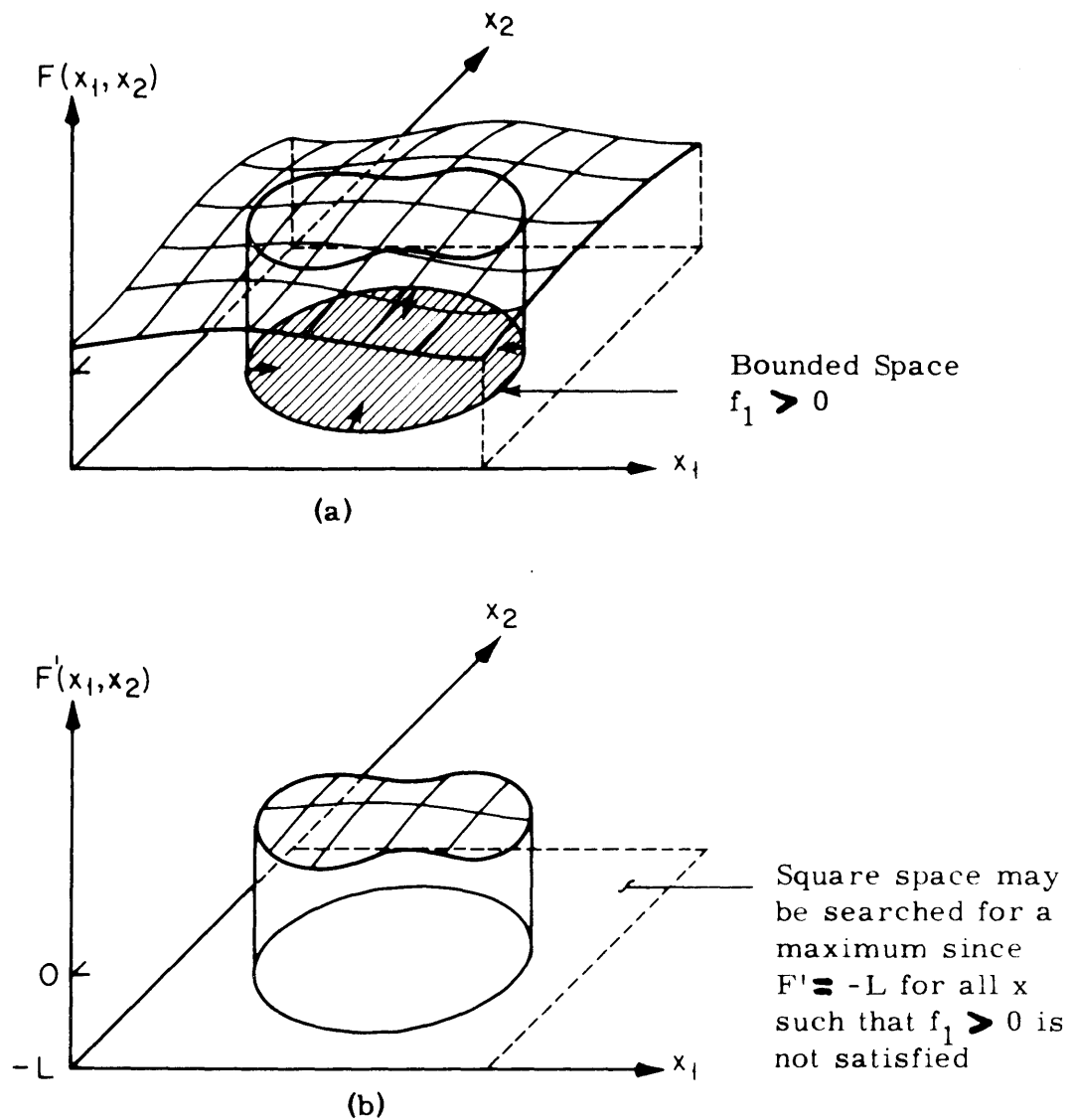


Figure 2-3

One Way to Search an Arbitrary Bounded Region

As discussed in the text, any bounded region can be contained in a larger hypercubical space, and this larger space can then be searched. By means of some device such as that shown above, points which happen to lie outside of the region of interest must be rejected from consideration. If convenient, this rejection can be done before the function is evaluated in the interest of efficiency, but in other cases it is more convenient to let the function be evaluated even though its value will have no significance.

x is chosen which is outside the valid search region, the value of the function is given a low value, $-L$, independent of whatever value of F might be computed. In this manner the problem of finding the maximum of a function defined on a complex bounded region is transformed into the problem of finding the maximum of a discontinuous function defined on a larger but simpler space.

A final type of search space is shown in Figure 2-2(d). In this case some of the parameters take on only discrete values, while others take on continuous values. In the case shown the problem can clearly be thought of as 5 one-dimensional problems, and each one could be solved separately. However, the techniques to be developed do not require this, and there are good reasons for not breaking up the problem into separate parts.

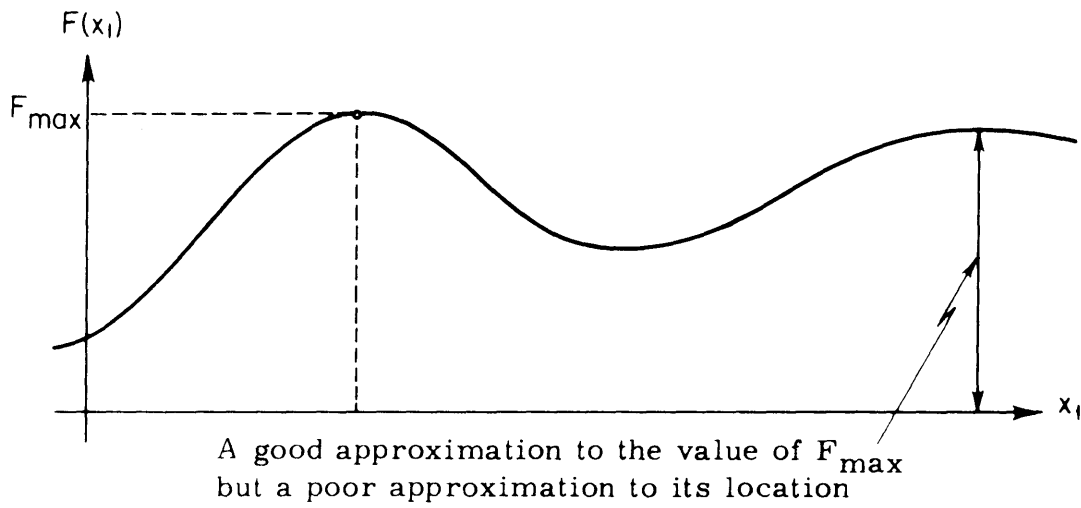
The spaces to be searched can be continuous, discrete, or mixed, and the valid regions for the search can be multiply connected as long as they are bounded somehow. The number of variables must be finite, and in practical problems only a space with rather a small number of dimensions can be searched with any thoroughness in a reasonable time. Since it is very uncommon in engineering to search interesting functions of more than two variables, even a modest increase in the number of dimensions searched should be important. This does mean, however, that any problem involving functionals, i. e., functions of an infinite number of variables, must be artfully reduced to a problem involving only a function. This is precisely what is done in Chapter 3.

There still remain some search or optimization problems which are too special to be discussed here. An example might be the case in which the parameters were related to each other by means of an implicit relation $g(x_1, x_2, \dots, x_n) = 0$ which could not be cast into the form $x_1 = g'(x_2, x_3, \dots, x_n)$. This would require first a search for a surface in n -space, and then a search along the surface for the maximum of the function.

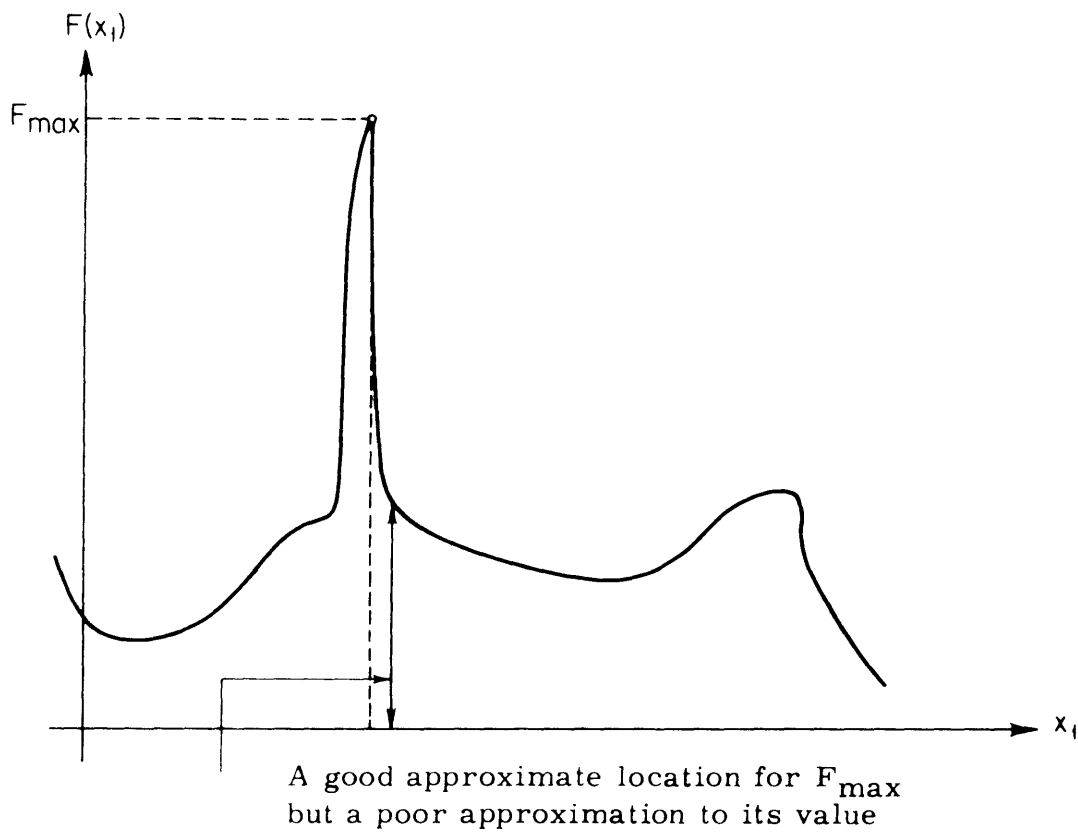
2/2/2 Location and Value

There are two aspects of a maximization problem which may be of interest. One aspect involves the location in x which produces the maximum of F , F_{\max} , and the other aspect involves the value of F_{\max} . In some cases F_{\max} can be uniquely associated with a single location. This may occur if the x space is discrete or if infinite precision were possible in a continuous case. However, it is also possible that a very good approximation to F_{\max} can be obtained at a location very far from the location of F_{\max} . This situation is shown in Figure 2-4(a). Figure 2-4(b) illustrates the opposite case in which locations near the location which produces F_{\max} do not produce values of F which are near F_{\max} . This illustrates the fact that to some extent these two aspects of a maximization problem must be considered separately.

Depending on the problem, one aspect or the other may be of prime importance. In some optimization problems the value of F_{\max} may be of no significance because the criterion function merely expresses an



(a)



(b)

Figure 2-4

Two Examples Showing that Convergence in Location and Value for an Extreme Need not be Strongly Related

ordering of preference among alternatives. In another example, the search for zeros of a function, F_{\min} is known in advance, and only the location can be of interest.

In other cases only the value of the extreme of the function may be of interest. For example, the maximum possible efficiency of an ideal machine may serve as an upper limit to the efficiencies of all real machines, but the parameter settings of such an ideal machine may have no useful relationship to the parameters of real machines.

There are more interesting aspects to the question of location than value because of a generally many-to-one relationship between x locations and F values. For example, ridges or plateaus are regions in x for which $F(x) \cong F_{\max}$, and in the search for a maximum it is often of great importance to recognize such features of the function. A plateau indicates that the function is very insensitive to changes in x , and a ridge may suggest a similitude relation between some of the parameters.

2/2/3 The Role of Trial and Error in Problem Solving

Before examining the details of a theory for searching for the extreme of a function, it may be worthwhile to discuss the role of trial and error in the context of general problem solving. The idea of trial-and-error methods for solving problems is certainly an old one. An interesting historical review and discussion is presented in the article, (Campbell²).

As soon as any problem is thought of as a search problem, the notion of a trial-and-error procedure is natural. The words trial and error, however, seem to carry a connotation of random, chaotic, undirected behavior in many discussions. Even if this is the proper definition of trial and error, the one to be proposed here is considerably broader.

It would seem that any problem-solving operation can be thought of as a search. This is because the notion of a nontrivial problem implies the existence of various alternatives available to the problem solver. Certainly many problems are solved using incomplete, mental, or analytic searches, but they are searches nonetheless. In fact, the techniques which appear least like searches are merely specializations of the results of very general previous searches. From these general studies, the rigid rules or algorithms for finding the solution to a problem are derived.

For the time being, a Deterministic Method shall be defined as any method for solving a problem or conducting a search which relies on rules or algorithms which under identical conditions always would yield identical behavior. Several examples will be discussed in a subsequent section.

On the other hand, random methods are those which are not completely predictable, although much more can be said about the operation of some random processes than that they are chaotic and undirected.

In a random search procedure, or random scan as it shall often be called, the probability that any point in the space of alternatives will be tested is known. This is in contrast to a deterministic method or algorithm in which the results of calculations determine which particular point in x will be tried next. The argument here is that trial-and-error methods usually thought of as random are not necessarily crude and of little use in searching a large space. In fact, the question of random versus deterministic methods is badly posed.

All this is clear if two basic propositions can be granted.

(1) The class of all random search procedures includes the class of all deterministic search procedures.

(2) Essentially all search procedures (with only trivial exceptions) involve some sort of trial-and-error procedure to be numerically useful.

Neither of these propositions is really very startling, but their implications are important. The first proposition is a statement similar to one often made regarding randomized and nonrandomized decision functions to be discussed later. The notion that a probability of unity implies certainty and a probability of zero implies impossibility is required to see that a deterministic scheme for selecting a single trial point for the next trial can be regarded as a scheme for adjusting the probabilities of a random search. Deterministic schemes simply set the probability of a single point to be unity and all others to be zero.

Clearly then, it is not a question of random versus deterministic, but rather a question of how much the distribution of a random search should be influenced by calculation on past trial results. In a later section we will present general random methods which provide a continuous transition between deterministic and chaotic methods.

The second proposition requires a bit more effort to demonstrate. The point to be made is that even the simplest algorithms still require a sequence of trials and errors and decisions based on these trials and errors. An answer cannot in general be given in a useful numerical form without conducting a series of trials.

As a case in point, consider the following simple problem of finding a number x such that $ax + b = 0$ for given a, b . Except for special cases ($a = b = 0$; $a = 0$; $b \neq 0$) the general solution is $x = (a^{-1})(-b) = -b/a$. In a sense, this solution is merely symbolic just as the solution to the matrix equation $AX + B = 0$, $X = (A^{-1})(-B)$ is merely symbolic even if A^{-1} exists. To express $-b/a$ in, say, decimal form requires the division algorithm which involves a set of sequential decisions for finding the successive digits in a decimal expansion. For each digit the divisor must be subtracted a number of times from the dividend to determine how many times this can be done without finding a negative remainder. The decisions are so simple and systematic that they can be mechanized quite easily, which leads us to forget that a search is required for each of the digits. The decimal expansion will

converge to the exact number, $-b/a$, if the process is continued indefinitely. The algorithm is a rule for solving a succession of problems similar to that of finding the zero of a monotonic function of one variable. The next section contains a demonstration that search theory can be profitably applied to such problems.

In searching for a ten place solution to $ax + b = 0$, any number of alternative approaches could be considered. For example, ten digit numbers selected at random could be tested; (there are only a finite number of them after all). This is clearly a crude trial-and-error procedure equivalent to picking points in a ten-dimensional space of discrete variables, each of which takes on the values 0 through 9. The division algorithm succeeds in transforming the problem from 1 ten-dimensional search to 10 one-dimensional searches in analogy with Bellman's Dynamic Programming technique and gives a rule for conducting each one-dimensional search. That the one-dimensional searches involve trials, however, is undeniable and that the trials, under assumptions to be discussed later, could be carried out in a more efficient way using random techniques is also undeniable.

What can be said for this algorithm is that a starting point is given for all searches, a stopping rule is given, and an upper bound can be given for the number of operations required for a solution of specified accuracy. When more difficult problems are considered, even in the cases in which algorithms can be given, e. g. , $X = (A^{-1})(-B)$ the desirable properties given above are not so easily attained. Typically, what is first lost is an a priori

upper bound on the number of operations required to converge to a specified accuracy. It is clearly necessary that a process be convergent but not sufficient, for infinitely slow convergence is quite useless. The bewildering number of competing algorithms in cases like this is evidence that a method has not been found which is optimum in a wide and useful sense. These methods evidently involve much trial and error. If fast convergence does occur the trials have been selected in an intelligent way.

As the problems become more difficult, e. g. , general maximization problems, the algorithms may not produce convergent behavior even for an infinite process, and start and stop rules may disappear. In this case the trial-and-error nature of the algorithms becomes very evident, and randomness practically forces its way into them in order to answer embarrassing questions about where to start and what to do after the scheme has stalled.

The suggestion to be made here is that no scheme, if looked at closely enough, can possibly avoid using a trial-and-error approach, although there may be definite rules for deciding where to try next. The real argument is not that trial-and-error is not capable of solving hard problems, but rather that rules and heuristics must be discovered and patterns must be perceived so that great numbers of a priori possibilities can be demonstrated to be unlikely or impossible points for the solution as the trial-and-error process proceeds.

One of the main points to be made in this paper is that rather simple

random scan schemes can compete successfully against complex, heavily deterministic ones in the sort of problems under discussion. In many cases, no deterministic scheme can guarantee convergence, much less estimate a convergence rate. The typical failure of deterministic schemes to converge with any speed even to a relative maximum indicates that the premises on which they were based often are not useful. We shall develop random scans which have the requisite flexibility between the crudest equiprobable trial-and-error scan and the most rigid deterministic sequence to be truly efficient at attacking these difficult problems.

2/2/4 Decision Theoretic Aspects of Search

Problem Formulation

In this section various aspects of the search problem which have been studied in various fields under different names are discussed. For example, Decision Theory, Experimental Design, Game Theory, Sequential Decision-making, Information Theory, and as we shall see later, Perception Theory, all bear on the problem of search. See (Abramson, Bross, Chernoff, Flood, Goodman, L. A., Hurwicz^{1, 2}, Milnor, Minsky, Savage^{1, 2}, Selfridge, Truxal², Wald).

Perhaps the term "Decision Theory" is the most general of these, and a search is after all little more than a series of decisions about where to look next and when to stop looking.

The concern here shall be primarily with the case in which there is complete freedom of choice of trial values of x within the bounded search domain. At any stage of the search, then, the decision must be which value

of x to try next in order to find a value of $F(x)$ which is near to the extreme of F , or find a value of x near to the value of x which produces the extreme; (these may or may not be equivalent problems). In either case, these decisions should produce a search behavior which is optimum or nearly optimum in some defined sense.

It should be pointed out that not all search problems have such a freedom of choice for successive trials in x . For example, an airplane searching the ocean for a submarine must travel a continuous path over the surface. In this case the decision involves the direction of travel to be taken at any time, and the trials occur continuously rather than one at a time. In cases like this, the process could be described as a random walk which would include deterministic walks rather than a random scan, although the term random scan includes walks as a special case. In any case, however, we shall not attempt to discuss truly continuous motion of the trial point in x . No matter how small the increments in x may be from trial to trial, $F(x)$ will be computed separately at each trial point.

The decision problem, then, is as follows: After n trials the search scheme must decide which value of x to try next or whether to terminate the search. The most information that can be available is the remembered locations of the previous trials in x and the corresponding values of $F(x)$, plus any a priori knowledge about the function, plus a statement about the criterion which will be used to judge the search process itself.

One must consider that the criterion for judging a search procedure must include, however vaguely and qualitatively, some practical elements

to insure that expensive and complex methods are not used for problems which are of limited importance.

A formal exposition of the decision problem can be found in (Wald). When applied to the search for an extreme, Wald's "decision function" is no more than a rule for choosing the next value of x or stopping the search. Again a randomized decision function bears the same relation to a nonrandomized (deterministic) decision function as our random methods of search (scans) do to deterministic methods. In a formal exposition of the decision problem associated with a search problem, the complication introduced by the necessity of sampling F by choosing x values rather than sampling F directly is obvious. Since x and F are not related 1-1, both value and location must enter into Wald's functions, not just value as in the usual decision problem.

The question of limitation of the search by a proper formulation of the problem is an interesting one. It is clear that if one merely asks for the precise value of F_{\max} or its location in any case in which there is a continuous variable to be searched and at least a semi-continuous F , an infinite number of trials will be required.

Perhaps the simplest way to limit the search is to simply ask for the rule which will yield the best solution in some sense in a given number of trials. Indeed, one of the most attractive possible results would be a decision rule which would yield an optimum behavior for any given number of trials, j , and such that if a better approximation were desired by allowing $j+k$ trials, this could be attained simply by continuing the process k

more times after j trials had been run in a way that was optimum for j . This property, which is like that incorporated in Bellman's Principle of Optimality, cannot in general be assured in a given scheme.

A less direct way to limit the search is to charge a cost to experimentation. For example, if the problem is to estimate the maximum of F , the choice of a decision rule could be based on an estimate of the performance of various rules in maximizing $F^* - C$. Here C is a cost which is function of the trials in x and generally increases with the number of trials, and F^* is the largest value of F found during any sequence of trials. Clearly, after a time under any decision rule, the increasing cost will swamp out any possible increase in F^* and an effective rule must terminate the search.

The nature of the cost function is one of the main determinants of rational procedures used in various fields which attack problems that are otherwise very similar mathematically.

For example, the problems studied in the Statistical Design of Experiments are often associated with very high experimental cost. In agricultural or medical experiments there is often so much noise in the function that the experiment must be run repeatedly at the same parameter point, x , before one can say that a significant trial has been performed. In cases such as these the problem is to maximize an average of the function rather than the function itself. Even if the noise were not present, the individual trials or experiments are likely to be both

time consuming and expensive in other senses. The result of this high cost of experimentation is two-fold. In the first place, it is simply impossible to search a large x -space. Very often the space is merely a few discrete points. Secondly, almost any amount of analysis of the data can be done, since it is a relatively inexpensive way to get more information out of the available data. This means that the decision rule can be very complicated.

On the other hand, much of what will be discussed here will apply to cases in which the cost of trials is not particularly high. In the first place, those cases shall be treated in which the noise in the value of the function is not excessive. The search theory will be developed as if the evaluation of the function were perfect at first, and then the effect of relatively minor inaccuracies in the F values will be discussed.

Secondly, the cost of storing and operating on the past trial results as well as the cost of trials themselves will be considered. This has apparently been rarely, if ever, considered explicitly in the literature. There are cases, particularly involving the use of an analog computer, in which the cost of the trials is relatively low, and the cost of operations on past results is relatively high. Clearly, such a situation in general favors methods in which a large number of trials are run with only simple directions about where to locate each successive trial. As will be seen in succeeding sections, this simple rearrangement of

emphasis on costs leads to a large change in rational search procedures.

Strategy Problems

J. von Neuman has used the term strategy in much the way we have used the terms decision rule or decision function (von Neuman). A strategy in his sense is a rule for making decisions during the play of a game. In this section criteria for choosing among possible strategies or decision rules for search problems will be discussed.

There is at least a plausible connection between the von Neuman Game Theory and any search problem; for in a sense, the search problem can be regarded as a game against Nature, (Milnor). Given a function F , the search scheme is interested in approximating two properties of the function, i. e., the location and value of F_{\max} . In general, some knowledge can be assumed about the function before the search begins. For example, it may be assumed that the function is continuous, that F_{\max} is more likely to be found in some regions of the x space than in others, or that the function is monomodal. This a priori knowledge will not specify in any complete sense the two properties of the function which are of interest. This is the purpose of the search. It may be assumed then that Nature chooses the actual functional characteristics from among the possibilities left by the incompleteness of the a priori knowledge. In many cases, particularly if the space x is of more than two dimensions, it may not be possible to assume much at all about the characteristics of F .

If it is assumed that Nature is playing this sort of a game it is clear that an effective search strategy must depend on the strategy of Nature. If Nature can be relied upon to choose functions that are easy to search most of the time, e. g. , functions with a single broad peak for x within the search space, then it is reasonable to attempt to attain fast convergence by using a highly deterministic search rule based on the assumption that the function is indeed a simple one. If, on the contrary, Nature has actually chosen a difficult function, perhaps with discontinuities and several peaks, then this strategy will not be appropriate.

The fundamental problem in the theory of search for an extreme has now been reached. In fact, since much of human activity is a search of one kind or another, the problem is fundamental indeed. Is it true that "Nature is deep but not malicious"? Is Nature actively playing a game against us? If not, what can we assume about the manner in which Nature picks the problems for us to face?

The interesting philosophical problems raised by these questions cannot be discussed in general here, but within the specific context of search problems some answer must be given if a rational search is to be found. Historically there have been several attempts to settle these questions. Three approaches will be discussed here, but the interested reader can find further variants in the following references. (Chernoff, Goodman, L. A., Hurewicz^{1, 2}, Milnor, Savage^{1, 2}.)

Three statements that might be made about the strategy of Nature in choosing the functions to be searched are as follows:

(1) There might be absolutely no useful information about the choices of functions which Nature might make. If it is required to approximate the absolute maximum of a perfectly arbitrary function, it is clear that all procedures may well turn out to be equally good; that is to say, no good at all. (2) The possibilities open to Nature may be known and restricted. This can be important information, particularly if the possibilities are finite in number. (3) Known probabilities may be associated with the possible choices of Nature. While these probabilities may often be subjective and not rigorously definable, they are, nevertheless, often useful.

In the third case in which the probabilities associated with Nature's choice of various types of functions are known, it is clear that in principle a search rule might be found that would result in an optimum average performance. This is often called a Bayes' solution (see Bayes, Wald), and it is a very attractive one. The gradual buildup of experience in any field enables the practitioner to hypothesize these probabilities about the problem at hand, and thus attack problems with quite specialized methods which will probably work. Certainly if several functions have been searched which are believed to be similar to the function at hand, one might as well incorporate this information into the search procedure. On the other hand, it is hard to imagine a case in which a function which is not a member of a

small class of previously investigated functions could be known to have the a priori probabilities about properties useful for a Bayes' solution. For example, two algebraic functions with exactly the same structural form but parameters which seem only slightly different may well have their F_{\max} located at very different points in the x space. Hence, even functions which look alike in their mathematical form may not be alike in the rather singular characteristics in which we are interested, i. e., position and value of F_{\max} .

In the second case, in which the possibilities for Nature's choice but not probabilities are assumed known, several criteria for evaluating possible search strategies have been suggested. One strategy that is often associated with the name of LaPlace (Goodman, L. A., Todhunter) is to assume every possible choice for Nature to be equally likely, and then as in the Bayes' solution, to pick that decision rule which optimizes the average performance. One is not tempted to argue very strongly in favor of this criterion, for obviously in an application to a series of problems, the method may give far too much weight to remote possibilities so that the performance of the search scheme is almost always mediocre.

An alternative to the LaPlace criterion is the minimax criterion prominently used in game theory. The minimax criterion, in effect, assumes that Nature is actually malicious and will pick the function which is the most difficult for the searcher. Under this assumption, as in any

competitive game, a sensible rule for the first player is to choose a strategy that will be maximally effective against the most intelligent play of the opposition. Clearly, the opposition will try to minimize the effectiveness of the strategy of the first player -- hence the name minimax. Rigorous statements of the minimax criterion can be found in many references (von Neuman, Wald, etc.), but for the present only the general philosophy is important. Although the minimax criterion is clearly conservative or pessimistic in a game against Nature, since it leads one to assume the worst, still the criterion is quite rational. When a minimax solution can be found its performance against the worst Nature can provide is as good or better than any alternative, while its performance against any other choice for Nature is better than its performance against the worst. In this way, the minimax criterion provides a lower bound on performance. Other criteria will result in search schemes which may perform better or worse than the lower bound of the minimax solution, but it is just a matter of luck which will occur.

As a simple example of an attractive minimax search, consider the problem of searching for a zero of a function defined on the interval $0 \leq x_1 \leq 1$ which is known to be monotonic increasing. If on a trial value of x_1 , $F(x_1) > 0$, then the zero must be located to the left of the trial. If $F(x_1) < 0$, then the zero is to the right of the trial, and of course if $F(x_1) = 0$, the search is over.

If it is assumed that Nature could have chosen a function with a zero crossing anywhere on the interval $0, 1$ and the attempt is to narrow the range of possibilities for the location of the zero as quickly as possible assuming Nature will do her worst, the first trial value of x_1 must be at $x_1 = 1/2$. This is because if any value other than $1/2$ is chosen, Nature could have chosen a function with its zero in the larger of the two intervals to the right or the left of the trial, and hence in the worst case, the zero would have been located only within an interval of length $> 1/2$. On the other hand, a trial at $x_1 = 1/2$ will locate the zero in an interval of length $1/2$ no matter what function Nature has chosen. The succeeding search procedure is fairly obvious, for the same argument applies to the new problem of locating the zero on the new interval of length $1/2$ as it did to the interval $0, 1$. The procedure is shown in Figure 2-5.

This procedure is attractive for several reasons. First, it is a dichotomization procedure such as is advocated from an information theory standpoint. Secondly, the optimum minimax procedure for j trials can simply be continued to yield the optimum procedure for $j+k$ trials. The procedure converges, in x at least, very rapidly. The interval in which the zero must lie is of length $1/2^n$ for n trials, independent of the function. No other zero-finding procedure has this property. Appendix A shows how the usual decimal division algorithm can be interpreted as a trial-and-error search and how a more efficient minimax procedure can be devised.

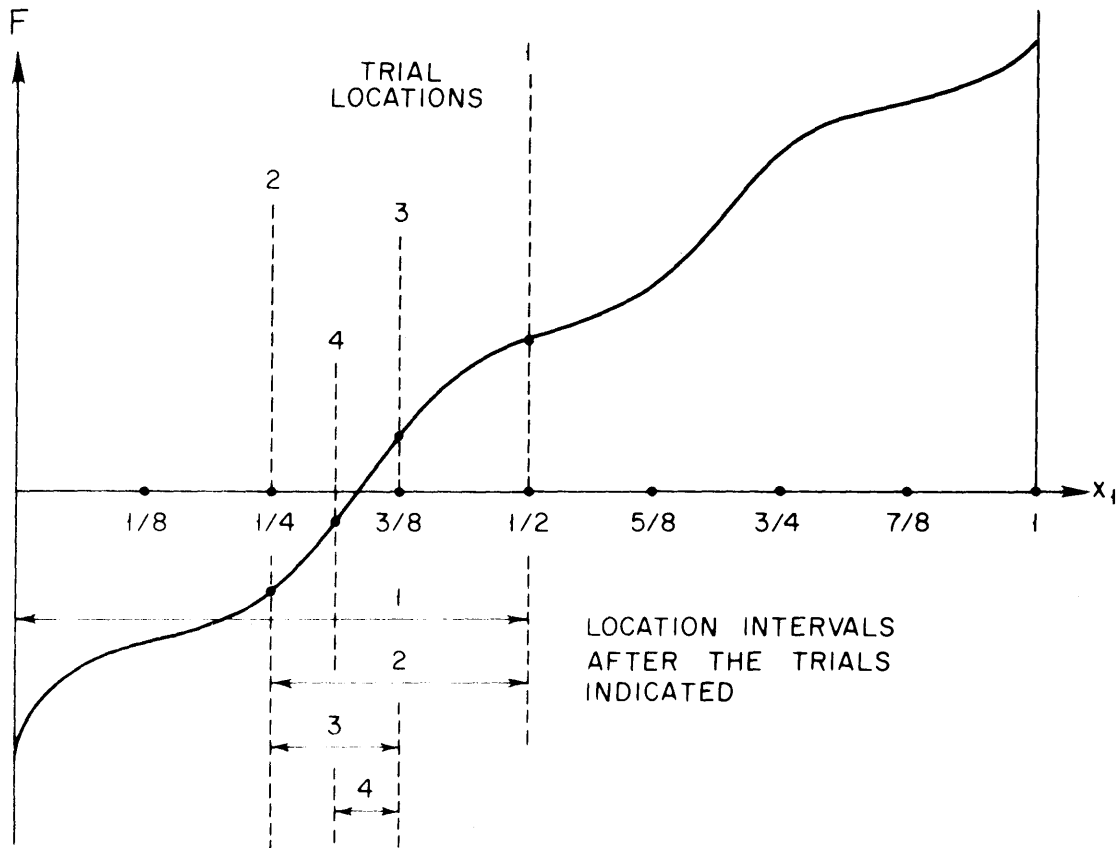


Figure 2-5

A Minimax Search for the Zero of a Monotonic
Increasing Function

The minimax procedure outlined above can be considered a very attractive search theory for finding the location of the minimum of some unimodal functions of one variable, since this can be equivalent to finding the zero of the derivatives of these functions. The procedure will even work in the case of discontinuous derivatives. Clearly, we would generalize these procedures to handle more complex functions of one variable, but practically the problem is not very interesting, since a rough graph of such functions will usually locate the peak quite closely. The search procedure outlined above is adequate for refining the accuracy of a graphical solution.

However, when the same logic is applied to a function of two or more variables, the beautiful minimax solution breaks down. As Kiefer has noted (Kiefer²), a single trial point in a two or more dimensional space, x , simply does not yield the same amount of information that a trial does in one dimension. This is not surprising if one considers that a trial point in a one-dimensional space dichotomizes the space, while it takes an infinite number of points (a line) to dichotomize a two-dimensional space, and in general, an $n-1$ dimensional subspace to dichotomize an n dimensional space. Hence using zero dimensional trial points the gain in information per trial cannot be nearly as great in a space of many dimensions as it is in a space of one dimension. In fact, as Kiefer points out it is not hard to prove in almost any case in which Nature's choices are not very heavily restricted that all search procedures are trivially

minimax in the sense that an infinite number of trials is required to make iron-clad guarantees about the accuracy of the approximations to F_{\max} and its location. This is not to say that some procedures will not be useful, but rather that if Nature desired to play against the search procedure, she could pick functions so difficult that no strategy would be any real good against them.

Obviously, if Nature's choices are restricted very heavily (as in linear programming, for example), there is a chance that a minimax strategy can be found. What is important about Kiefer's paper is that even in the simplest, most restricted cases the strategies seem "computationally much too messy" to be of practical interest. On the other hand, in an engineering context where a priori guarantees about how many peaks the function will have and about continuity and scaling (e.g., peak width) can rarely be made, the possibility for a useful minimax strategy seems remote.

If the minimax criterion is not useful, we can still try other criteria such as the Bayes' or LaPlace. But evidently we must be prepared for situations in which Nature can choose such difficult functions that the search strategy is still dominated by these possibilities even if their probability of occurrence is not large. The result may well be quite unsatisfactory, a search procedure which like all procedures does not work well in difficult cases, and yet because of an attempt to improve on the nearly hopeless ones, does not even work well in easy cases.

The most common historical approach, examples of which shall be seen later under the topic "Highly Deterministic Methods", seems to have been the common sense one of attacking all problems with tools that will work well in simple cases. This might be considered an implicit assumption that Nature is trying to be helpful, or more reasonably that it is not useful to waste much time on very hard problems while there are easier ones remaining.

It is a symptom of the basic difficulty of search problems that the discussions of convergence in most methods are of little value, since these discussions always start with restrictive hypotheses. Typically, the hypothesis is that a good approximate solution is available before convergence is discussed and that local parameters of the function in the region of the maximum have certain values. It will be the contention in the following pages that most of these procedures are only suited for the relatively simple problem of refining fairly good approximate solutions and are not suited to the initial stages of a search. The random search methods to be proposed will still work best on simpler cases and not well on the most difficult ones. However, the methods will be quite different from those proposed historically and will be much more suited to true search problems rather than just refinement problems.

The Role of Experiment

While experience may be very important in classes of search problems, generalization about the effectiveness of search procedures based

on experimental studies may be very dangerous. In the problem of finding the maximum value of a number of functions of several variables which are all closely related, any search procedure which has been effective on most of the functions may be expected to work fairly well on others. This is the philosophy at least of a Bayes' solution.

Experimental studies have been made which tested the performance of various search rules on a few functions (Brooks¹). If as was suggested these functions were typical in some sense of those found in practice, the implication would be that the effectiveness of the various schemes had been measured.

In light of the philosophical difficulties outlined above, however, it would seem very dangerous to make any strong generalization from the experimental data. First of all, it is hard to imagine a rigorous sense in which as few as three functions could be shown to be typical of the infinite number of functions which could occur in practice. For example, does a test on a function of two variables provide any insights to functions of three or more variables? It has already been shown that results for a one-dimensional function do not generalize to functions of two or more variables. It also seems at least possible that test results on three other functions which might look similar to the functions used could lead to quite different results. Indeed, even starting the procedures from different points on the same function clearly might have the same effect.

In the tests performed each function was merely a single broad peak. Given this preknowledge, the parameters (such as a step size) of almost any reasonable search rule can be adjusted so that good behavior will result. In practice, however, the setting of the parameters of a search or decision rule is often mere guesswork, so that getting the scheme to converge is usually a matter of some trial and error.

In the Brooks experiments the crudest equiprobable random scan was the only random method tried, and surprisingly, it performed quite well. But some simple modifications of the random method will be shown which with the available preknowledge about the functions would allow the random methods to be adjusted for much superior performance. Alternatively, increasing the complexity of the function would not affect the behavior of the crude random method very much (quite possibly not at all), but would seriously affect the other methods tested.

The reason that there are so many search rules offered in the literature is that it takes an immense amount of data to prove that one method is generally superior to another experimentally. Without extensive data one is led to perceive differences which do not exist. For example, Brooks compares a pure random scan in which say k^2 points are picked in x with uniform probability for any point in the square $0 \leq x_1 \leq 1$, $0 \leq x_2 \leq 1$, with what he calls a stratified random scan. In the latter case he divides the square with a $k \times k$ grid and then picks a point at random within each subsquare. See Figure 2-6.

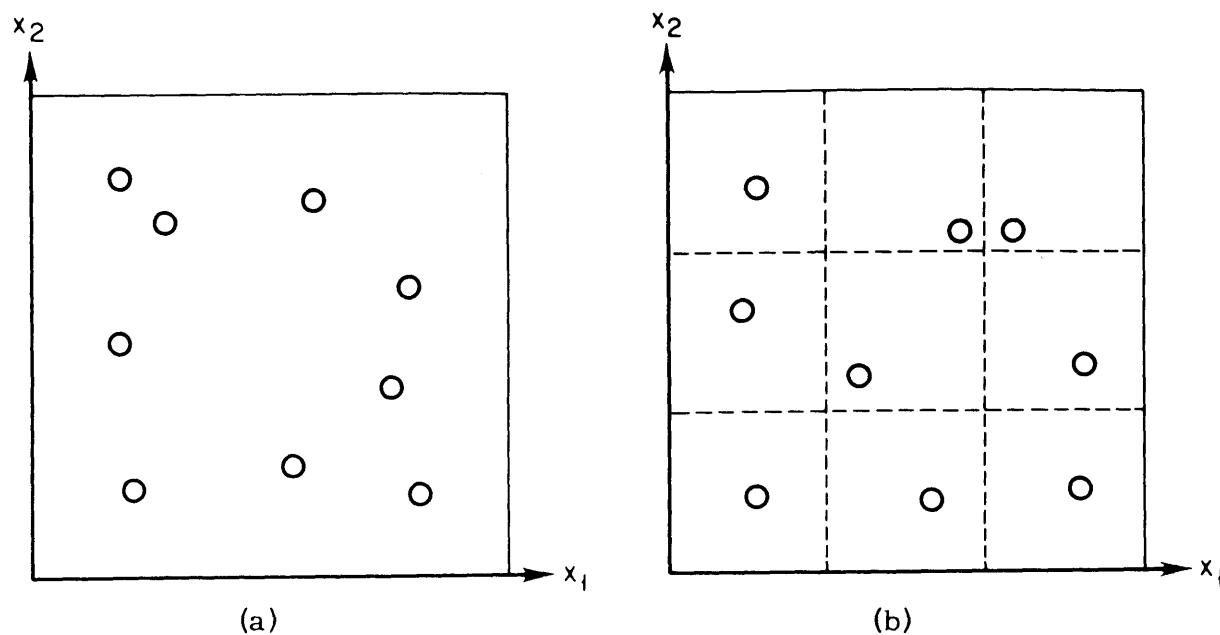


Figure 2-6

A Comparison Between Pure Random and Stratified Random Scans

In (a) nine points have been chosen at random using an equiprobable distribution. In (b), the stratified random scan is used. In each of the nine subsquares of (b) one point is chosen again using an equiprobable scan. The idea of stratification can be generalized in a number of fairly obvious ways. It is probably not worth the trouble except in those cases in which only a small number of trials can be made.

As we shall show later, it is easy to predict the average greatest height of a function attained on a series of trials performed according to these two methods. Such an average performance depends entirely on the distribution of trials in the x space which is identical in these two cases. Hence the average performance will be identical, yet an experimental difference was reported. This is not so surprising, since only a few runs were made with each method, and it is true that the variance not the mean of the highest value attained is different for the two methods. A statement that the stratified random procedure guards against certain bad distributions of points in x as well as preventing certain unusually fortunate ones would be of interest. But to report different mean accomplishments is merely to point out an inadequacy in the amount of data taken.

The random methods of search to be proposed here can be tested and, most importantly, understood by means of mental or mathematical experiments. Hence it was felt to be unwise to conduct extensive computational experimentation. Instead, we propose to demonstrate that the methods are novel, useful, easily achieved in actual computational machinery, and flexible enough to be suited to the extremely singular features of particular cases.

2/2/5 Highly Deterministic Methods

It has been pointed out that the class of all random search procedures can be claimed to encompass all possible methods. However, in the past, search procedures were commonly given in the form of deterministic algorithms with perhaps only the starting points to be determined by a guess

or some other random method. In this section the characteristics and limitations of such methods will be discussed. Since the number of such methods, or variants of methods, is very large, only the general features of these methods will be discussed. The interested reader can find expositions of the details of some of these methods in the following references: (Anderson, Booth, Box, Brooks¹, Brown, Crockett, Curry).

Perhaps the most common characteristic of these methods is their basis in differential geometry. Since a search problem is by definition a problem of geometry in the large rather than in the small, all the methods to be discussed here have rather similar limitations and characteristic failings.

A large class of methods is based on the notion that if the function F is continuous, one can evaluate the change in value of the function, δF due to an infinitesimal change in location x , i. e., a change from the point x_1, x_2, \dots, x_n to a point $x_1 + \delta x_1, x_2 + \delta x_2, \dots, x_n + \delta x_n$, if the first partial derivatives of F exist. $\delta F = \frac{\partial F}{\partial x_1} \delta x_1 + \frac{\partial F}{\partial x_2} \delta x_2 + \dots + \frac{\partial F}{\partial x_n} \delta x_n$.

Clearly, if the partial derivatives do exist and if one could compute them, it would be possible to find an infinitesimally larger or smaller value of F by making an appropriate differential change in x . Strictly speaking, however, this is of no use computationally, since only finite changes in x and F are of interest. Thus the idea is modified to the following:

$$\Delta F \stackrel{?}{=} \frac{\partial F}{\partial x_1} \Delta x_1 + \frac{\partial F}{\partial x_2} \Delta x_2 + \dots + \frac{\partial F}{\partial x_n} \Delta x_n$$

where the Δx_j 's indicate finite changes. The equation is presumed to be approximately correct for small Δx_j , but it is not generally known definitely just how small the Δx_j 's must be. (This depends on the magnitudes of higher derivatives which are not usually known.) Also the derivatives must be computed using small but finite increments, and the values obtained will be good approximations if the increments are really small, and bad approximations if the increments are really large. Another problem is that if the increments used to evaluate the partial derivatives are too small, the operation will involve subtracting values of the function which are very nearly equal, creating a bad accuracy and noise amplification problem. For this reason, the method requires a fairly good estimate of appropriate increment size.

The final step in the cycle is to move to a new value of x and begin over again. There are many ways to make this final step. One way is to use the evaluated partial derivatives to estimate the direction in x which presumably will result in the greatest change in F , as in the Steepest Ascent or Descent Methods. Here another compromise is needed, since a very small step in the indicated direction will result in only a correspondingly small change in F , while too large a step may not result in an improvement at all.

In order to have the scheme converge, the steps in x are usually made proportional to the magnitude of the greatest slope or gradient. Then if the scheme approaches the top of a hill, its steps will decrease in

size until the scheme converges to the top. The scheme will also slow down and stop at any place which appears locally flat, a plateau, a saddle point, or a minimum, although in some cases the system may eventually diverge and escape. When such a scheme does converge it will be to any relative peak, and if an absolute maximum is required, the system must be set off again in another region of the space.

The convergence of such methods to precisely the top of a smooth peak from a point not too far from the top can often be quite spectacular. This is why such methods shall be referred to as refinement techniques rather than search techniques. The convergence to the top of a ridge-like peak may not be very rapid, and of course convergence to the top of the absolute peak is a matter of luck in the starting point if there are many relative peaks.

An important limitation to the effectiveness of such a method is that discontinuities in the function will be highly disturbing, as will the presence of rather small amounts of noise. The latter property may be quite important in some situations in which the noise in the function cannot be eliminated.

Another class of methods can be based on the differential properties of smooth peaks by expanding F about the location of the critical points, i. e., the points at which the first derivatives of the function vanish (Morse). For infinitesimal distances from the critical point, the function, F , can be expressed by a quadratic form if derivatives up to the third exist. Hence

one can attempt to fit a quadratic surface through a sufficient number of trial points adjacent to the peak in order to predict the location of F_{\max} . It should be evident that all such curve fitting procedures can be used for refinement of a good approximate solution or for any function which happens to have a shape nearly quadratic in the large. As a general search rule, however, this sort of curve or surface fitting is very poor because the surface is fitted only locally, and predictions of the functional behavior far from the local region may be so bad as to be useless. Once again this is a type of scheme which is useful for final convergence or refinement but not for an initial search.

It would seem from a search of the literature that most of the formal methods proposed for finding the maximum or minimum of a function are really useful only in the case of particularly simple functions or for the refinement of a good approximate solution. All seem to be sensitive to a lack of continuity in the function, or to noise or inaccuracy in the evaluation of the function. To use these methods for general problems, they must be combined with some sort of true search procedure. One of these combinations will be discussed in a comparison with our proposed random methods.

2/2/6 General Random Methods

A Hierarchy of Random Scan Schemes

In discussing a random scan of a search space, x , a probability density function or distribution function must be defined to express the

probability that any point in the space of alternatives will be chosen by the chance mechanism. The words "pure random scan" denote a process of selecting points in the search space, x , according to a probability density function which does not change from trial to trial. Perhaps the most common density function used in this regard is the function which leads to an equiprobable distribution of points in x , but it is not the only one. The equiprobable scan is often the easiest one to generate on a computer, and may represent a conscious use of the LaPlace criterion mentioned earlier.

When the bounds on the search space are given as part of the problem, it will be efficient to have the probability density of the scan drop to zero for all points outside the boundary. However, if the bounds on x merely represent an a priori belief that the desired solution will be in a given region, it may be better to strongly deemphasize these ill-defined boundaries and merely to have the probability fade gradually to near zero for points far from the expected solution region.

For the case in which all the parameters $x = x_1, x_2, \dots, x_n$ have a continuous range of variation, the scan probability density function is defined as follows: $s(x)$ the probability that the trial point will lie within the region $x_1 \pm dx_1, x_2 \pm dx_2, \dots, x_n \pm dx_n$ divided by the volume element $dx_1 dx_2 \dots dx_n$.

The usual conventions will be allowed to handle the cases in which there is a finite probability associated with a single point, e. g., if the x space is discrete or if a deterministic method is used to determine precisely which point shall be tried next. See Figure 2-7 for some examples

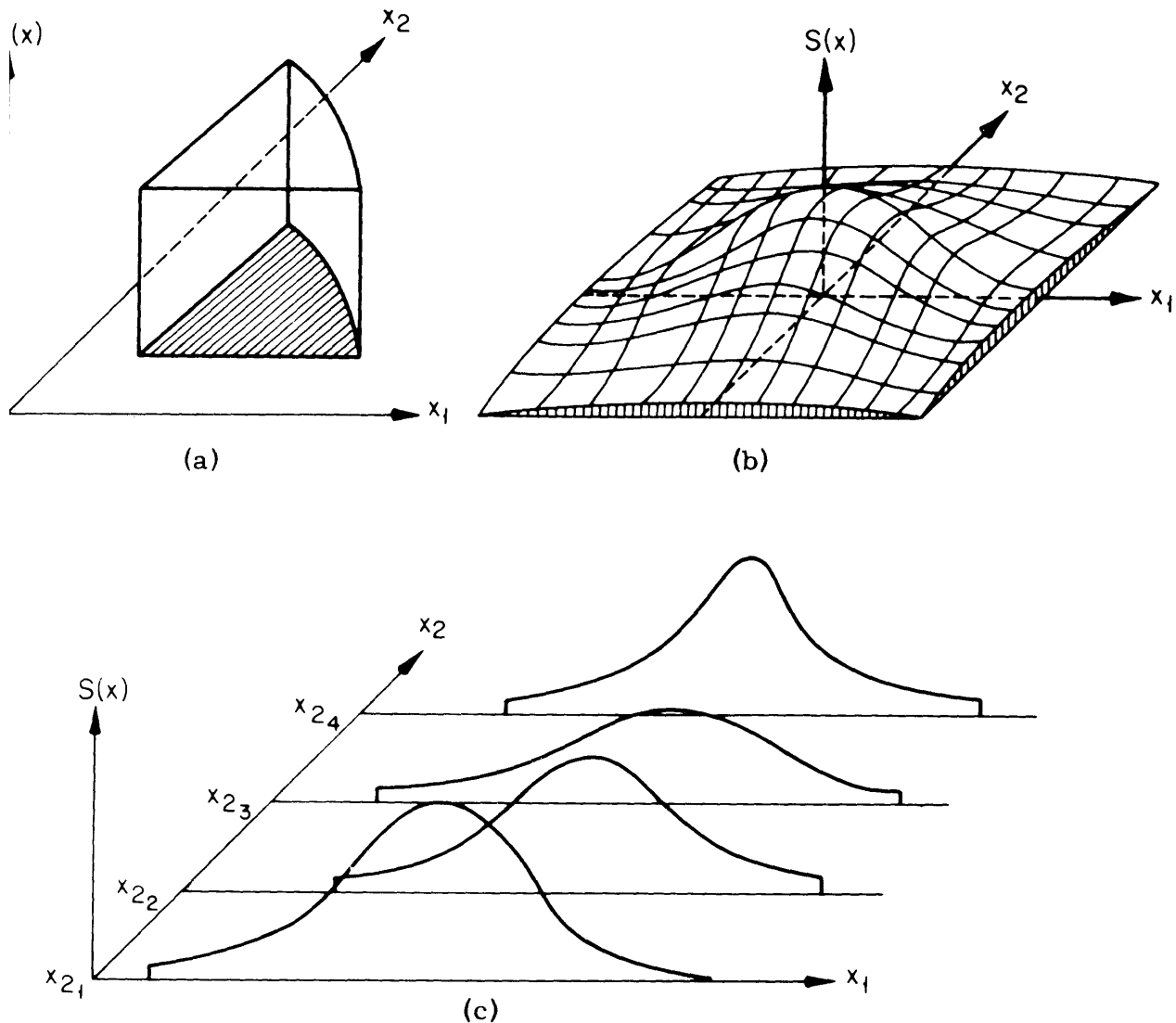


Figure 2-7

Three Examples of Two Dimensional Scan Probability
Density Functions, $s(x)$

- (a) An equiprobable scan over a bounded region
- (b) A scan with de-emphasized boundaries for searching primarily near $x_1 = x_2 = 0$
- (c) A scan for searching a discrete-continuous space. Note here that it is the sum of the area under all four curves that is equal to unity, not the area under one curve as would be the case for a one-dimensional distribution.

of $s(x)$.

The simplest search method based on a pure random scan involves recording the location and value of the largest value of F discovered after a number of trials have been made. We shall consider primarily the case in which the trials cannot be run simultaneously but rather must be ordered in time. In this case the storage problem is not acute, since after each trial the value of F just obtained may be compared with the previously obtained highest value, and if the new value is higher, its value and location will replace the stored value and location; if it is not, the trial results can simply be discarded. If two values of F are indistinguishable, it will not matter which location is stored as long as only a pure random scan is used subsequently. In this way using only enough storage to locate a point in the x space and a value of F , the highest value of F obtained on any number of trial evaluations of the function can be recorded. The success of this method clearly depends on the comparability properties of $1/1/1$.

It is clear that in general, the most important information to be stored during any search is the best trial value and its location. In the cases done sequentially in time, this storage requirement has the important property of being independent of the number of trials. For this reason, the schemes proposed shall always store at least this information. Of course the most information that can possibly be stored is the location and value of F obtained on every trial. This may be a rational procedure for small search spaces in which the evaluation of F is so noisy that

individual comparisons have little value, and a statistical study must be made after many trials have been run. In the cases of interest here, however, noise in F is assumed not to be an overwhelming problem, and the storage of information proportional to the number of trials is clearly inefficient in a general sense. See (Chichinadze) for a study of methods which do remember all this information.

A pure random scan search is powerful but not necessarily efficient. It is powerful since it can be arranged to conduct a trial arbitrarily close to any point in x including the precise location of F_{\max} simply by letting the number of trials grow sufficiently large. In the case of either computed functions or physical experiments, then, the scheme will eventually try the location which will result in $F = F_{\max}$, at least within the accuracy limits of the problem. It is clear, however, that after a large value of F has been recorded, almost all subsequent trials will be failures because the probability of finding a still higher value of F will be small. For this reason, final convergence of a pure random scan scheme may be quite slow.

The method also suffers terribly from, in Bellman's phrase, "the curse of dimensionality" (Bellman¹). Consider an n -dimensional space where x_1, x_2, \dots, x_n all have a bounded, continuous range of variation which for the purposes of this discussion can be normalized to 0, 1.

The number of trial points required under an equiprobable pure random scan such that points have been tried of the order of Δ from the

location of F_{\max} is $m = \Delta^{1/n}$. For example, if $\Delta = 1/10$ then $m = 10^{1/n}$. This can be referred to as a 10% scan; i.e., a scan such that points have been tried at an average separation of 10% of the full range of each of the variable parameters, x_i . The same type of behavior evidently occurs under any pure random scan, for example, a fixed grid scan or a scan such as in Figure 2-7(b) with fairly obvious modifications. Also the space clearly need not be hypercubical but may be bounded in other ways as long as there are not qualitatively different ranges for some of the variables. The discrete-continuous case of Figure 2-7(c) really degenerates to a series of one-dimensional scans in this calculation.

It is of interest to explain why the search problem should be so drastically affected by the dimension of the x space. As has been noted by von Neuman and others, the convergence of some random methods in the Monte Carlo techniques is, in a sense, not affected by the dimensionality of the problem, (Kac, Milne, Meyer).

In the search problem the region of interest is the zero-dimensional region associated with the precise location of F_{\max} which is imbedded in an n -dimensional space. In the Monte Carlo problems the region of interest is an n -dimensional subspace of the n -dimensional search space. For example, a Monte Carlo method of estimating the value of an integral $I = \int_a^b y \, dx$, with $y = f(x)$ is to pick y, x by an equiprobable scan and to test each choice to see whether the chosen y, x satisfy $y \leq f(x)$ or not. The approximate value of I will be given by the ratio of the number of points which satisfy the relation to the total number of points tried per unit area. See

Figure 2-8.

In the two-dimensional problem illustrated what is desired is an area ratio; in a higher dimensional problem it would be a generalized volume ratio. In any case, the dimensionality of the region of interest is the same as the dimensionality of the problem, and the convergence of the approximation behaves in a similar way for problems in all dimensions. In the extremization problem the dimensionality of the problem drastically affects the number of trials required in a fundamentally unavoidable way so that even for modest n problems become truly impossible to solve if the functions involved require much searching at all.

Obviously, what is required in order to search efficiently is to narrow the search space as the search progresses. In the one-dimensional minimax search example cited previously, the search space could be cut in half after every trial. If anything like this were possible in general, then very complex problems could be solved. This cannot be done generally in more than one dimension, even for rather restricted classes of functions. However, a narrowing of the search can be attempted, and by luck it may often prove effective. This is just what is attempted in the deterministic methods discussed above, and their occasional failure to converge is just a demonstration that infallibility cannot be guaranteed.

Random methods have the flexibility to introduce any amount of deterministic search narrowing into a search scheme by altering the scan distribution, $s(x)$, in very simple ways. For example, if the function, F ,

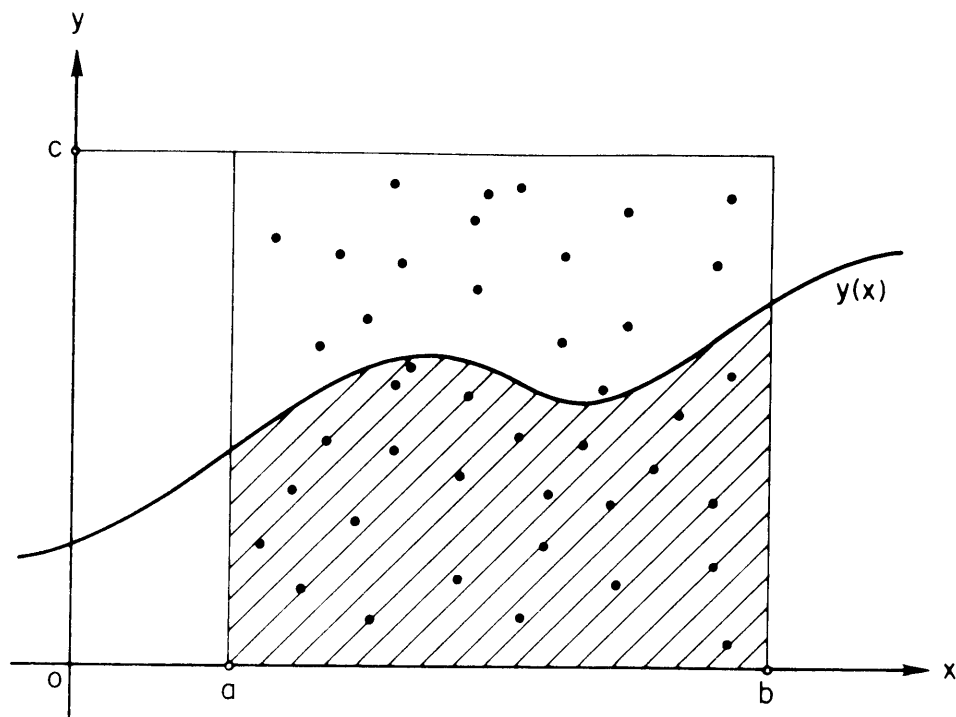


Figure 2-8

An Example of a Monte Carlo Method Applied to the
Evaluation of a Definite Integral

$$\frac{\text{The number of points under } y(x)}{\text{Total number of points}} \approx \frac{\int_a^b y(x) dx}{(b-a)c}$$

$$\text{or } \int_a^b y(x) dx \approx \frac{\text{Points under } y(x)}{\text{Points/Unit area}}$$

is smooth, then in the vicinity of the stored location of the best trial value, x^* , there will be points which are better (higher or lower). Hence a narrow scan centered at the location of the current best trial value will rather quickly find a somewhat better value. If the mean of the narrow scan is then moved to the new best trial value location, the process will repeat until a relative extreme has been found. Thus without computing gradients a steepest ascent type of behavior can be simulated. Experiments on an analog computer have confirmed that the narrower the scan, the more nearly a steepest ascent path is followed. See Figure 2-9.

Clearly, this procedure if used alone will generate frequent small improvements to the best trial value until a relative peak is reached, at which time the process will stall just as any gradient method would.

An interesting parameter of this procedure is the size of the scan, for as it is varied from very small to nearly as large as the entire search region, the procedure is modulated from a highly deterministic gradient method to a pure random scan. This sort of flexibility has apparently not been utilized in previous search schemes.

This type of scan, in its simplest form, has been discussed briefly in (Brooks², Favreau) and in the present author's earliest memoranda, and it is interesting to note that the analogy of this narrow scan with a moving mean to the process of evolution is so immediate that all three authors gave independent mention of it.

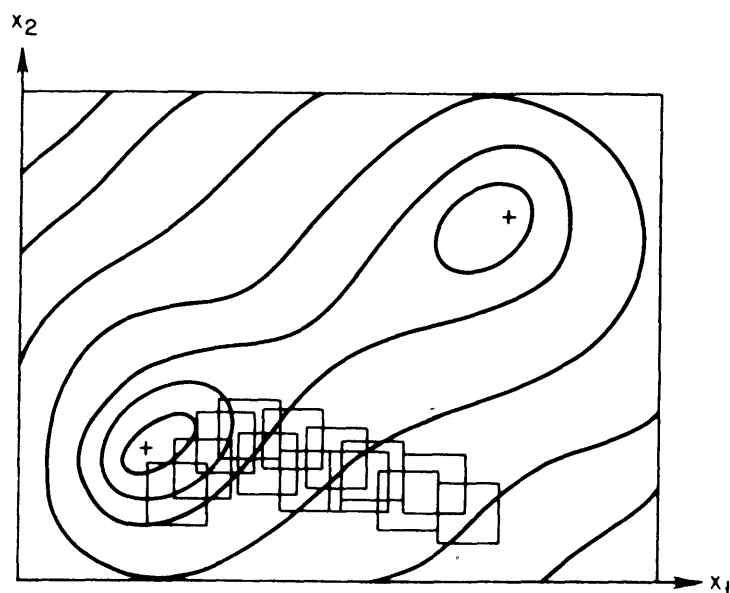


Figure 2-9

A Sketch of the Steepest Ascent Type of Behavior
of a Narrow Range Moving Mean Scan

The position of successive scans is indicated by the small boxes. Each scan is centered over x^* , the location which produced the highest value of F of all trials carried out up to any given time. Whenever a higher value of F is found, the mean or center of the scan is immediately moved to the new x^* .

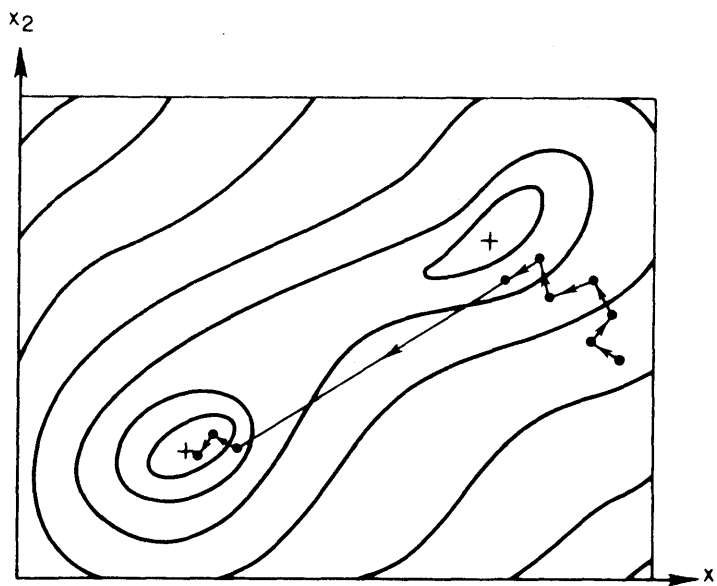


Figure 2-10

A Sketch of the Behavior of a Composite Narrow and Wide range scan

The scheme starts to converge toward a relative peak, due to the narrow, moving mean scan but eventually jumps to the absolute peak due to the wide scan. The final convergence to the absolute maximum is carried out in a steepest ascent manner. The dots indicate successive positions of x^* .

Another possibility is to use a narrow moving mean scan simultaneously with a pure random scan. That is, $s(x) = as_1(x) + (1-a)s_2(x, x^*)$ where $a, 0 \leq a \leq 1$ is a modulating constant, $s_1(x)$ is a pure random scan distribution, and $s_2(x, x^*)$ is a narrow scan centered at x^* , the location of the current best trial value of F . For " a " near unity, the scan is nearly a pure random one; for " a " near zero, the scan is mainly a narrow scan with a moving mean. This combination has the attractive feature of combining a wide search and small-step refinements. See Figure 2-10 for a typical trace of the stored best trial location, x^* , under this sort of scan. Ideally, the wide scan prevents a hanging up on a local peak while the narrow scan refines the solution to the peak nearest x^* , on the chance, as it were, that it might be the absolute peak.

Once the idea of introducing a determinism into a random scan scheme is grasped, it is an exercise in imagination to construct schemes which might prove useful in certain applications. For example, if two distinct peaks are expected in the search space, two narrow, moving mean scans could be used. One scan would be centered at the best trial location, the other centered at the location of the best of all trials observed at a distance from the stored best trial greater than some reference distance. After every trial two comparisons would be made; the first comparison to determine whether the new trial resulted in a value of F greater than the current stored highest trial value, and the second to determine (1) whether the new trial was far enough from the stored best trial location, and if so,

(2) whether the trial value was higher than the value for such locations previously tried. In this way, two peaks could be climbed at once as long as the reference distance was chosen properly.

Another modification to speed up the steepest ascent type of behavior would be to move the narrow scan not just to the new location of best trial value, but further on in the direction of the change in x^* . In fact, various shapes for the narrow scan could be considered which would express roughly, at least, a more nearly optimum gradient climbing procedure. See Figure 2-11.

It should be obvious that there can be no rule for constructing the fan-shaped search of Figure 2-11 unless a great deal is known about the function, just as there is no possible general rule for step size in any gradient method. The difficulty is clearly less critical in the random method, however, since instead of picking a single point to try and moving the search to that point, the search has been smeared out into a likely region, and the scan is only moved when improvement has been attained. The system might well converge satisfactorily even if it were not adjusted very well or if some parts of the function have very different local character than others.

While there are many other ways to introduce the traditional type of determinism into random procedures, we shall go on to discuss the convergence aspects of some of the simpler schemes. In later sections more interesting ways of introducing determinism into the search will be discussed.

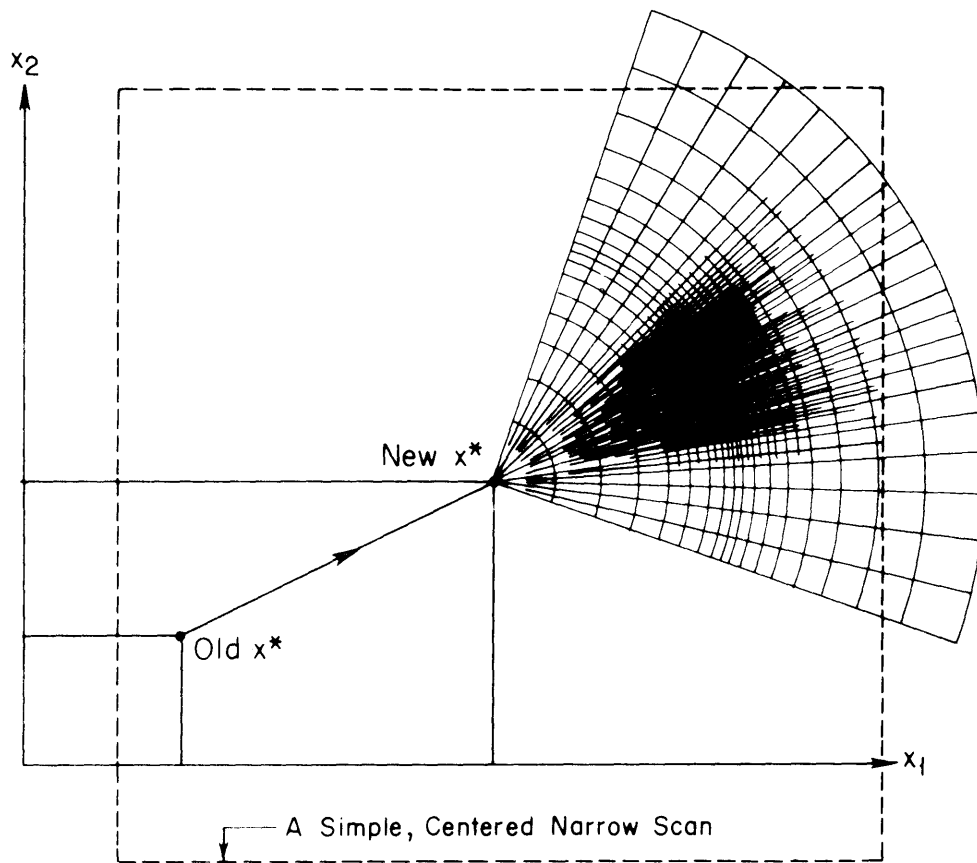


Figure 2-11

An Attempt to Optimize the Steepest Ascent Aspect
of a Narrow, Moving Scan

The fan shaped scan searches ahead of the best trial location, x^* , in the direction of the last improving change. The density of the shading is meant to suggest the density of the search. Obviously the optimum distribution depends on the surface to be searched and even on the particular locality of the surface being searched, but the general features of the optimum distribution will be similar to those in the sketch.

These methods can not be used in the older deterministic schemes. The main thing to note at this point is that the random schemes have a flexibility which allows the introduction of however much determinism is justified either by a priori knowledge or by experimental results.

Value Convergence

In this section attention will be turned to the study of the convergence of the value obtained on the best trial, F^* , to F_{\max} as a function of the number of trials, n , for several of the simpler random scans.

The case of a pure random scan is particularly simple and instructive. Given a probability density function $s(x)$ expressing the probability of choice of any point in the search space, x , which does not change from trial to trial, another distribution can be generated, the distribution of trial values of $F(x)$. If $P(F)$ is defined to be the probability that on a trial a value of the function will be obtained which is less than or equal to F , the operation of finding $P(F)$ given $s(x)$ and $F(x)$ may be indicated as follows:

$$P(F) = \iiint \dots \int s(x_1, x_2, \dots, x_n) dx_1 dx_2 \dots dx_n.$$

All x such
that the
value of the
function $\leq F$

In one dimension it may be possible to invert the function $F = F(x_1)$ to find $x = x(F)$, but in two or more dimensions this is not possible, since there is, in general, an entire n -dimensional subspace for which $F(x)$ is less than a given value. Indeed, if one could really find $P(F)$ from $F(x)$

by means of analytic operations, there would be no need for a search. It is quite easy to imagine the operations involved in the definition of $P(F)$ for any function which can be visualized. See Figure 2-12.

The distribution function $P(F)$ can be defined for any function, F , and scan distribution, s , in principle and always has the usual properties of distribution functions, i. e., $P(F)$ is defined the entire range of F , F_{\min} to F_{\max} (which could be $-\infty$ to $+\infty$) and is monotonic increasing from zero at F_{\min} to unity at F_{\max} , but it need not be continuous. There will be a jump in $P(F)$, for example, if a plateau exists.

For a maximum seeking search the distribution $P_n(F^*)$, the distribution of the highest value of F obtained in n trials, can be studied. A similar analysis can be made for minimum seeking searches. Now, since $s(x)$ does not vary from trial to trial, the series of trial values of F will all be independently and identically distributed according to the distribution function $P(F)$. The probability, $P_n(F^*)$, that after n trials the largest trial value will have been less than a particular value F^* is simply given by $(P(F^*))^n$. Figure 2-13 indicates the manner in which $P_n(F^*)$ varies with n .

Several features are immediately apparent from Figure 2-13. One is that the probability of F^* being less than the median value of F (the value of F such that $P(F) = 1/2$) diminishes very rapidly with n , since the n^{th} power of numbers less than $1/2$ approaches zero very quickly. However, near F_{\max} the probabilities are given by the n^{th} power of numbers near unity so that after n trials, it may still be probable that

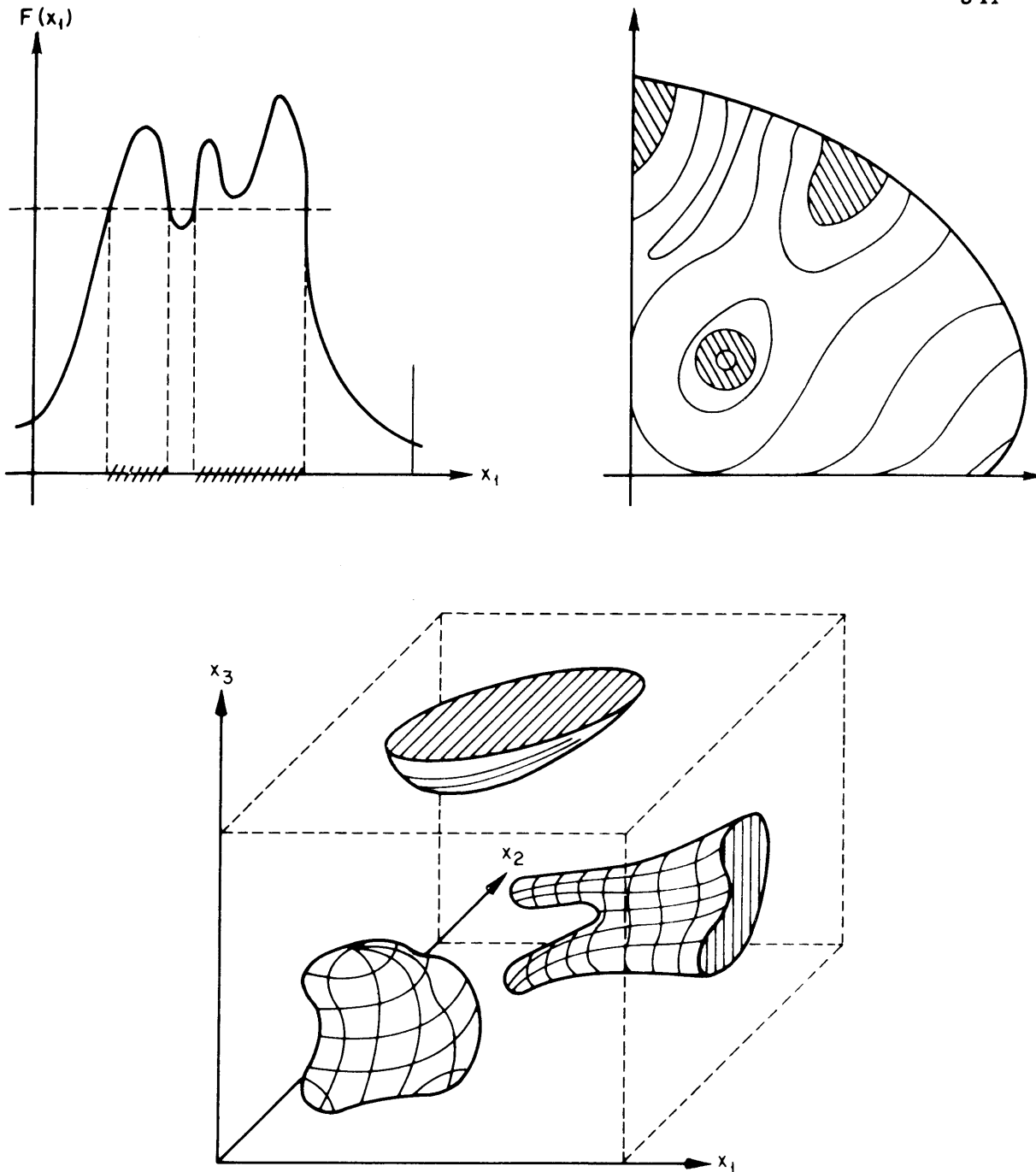


Figure 2-12

Sketches Showing How $P(F)$ is Defined for Functions of One, Two, and Three Variables

In each case, the shaded regions in the x -space are regions for which the associated function value is greater than a particular value, F . Therefore, $P(F)$ is the total probability given a scan distribution in x of a trial point falling in an unshaded region.

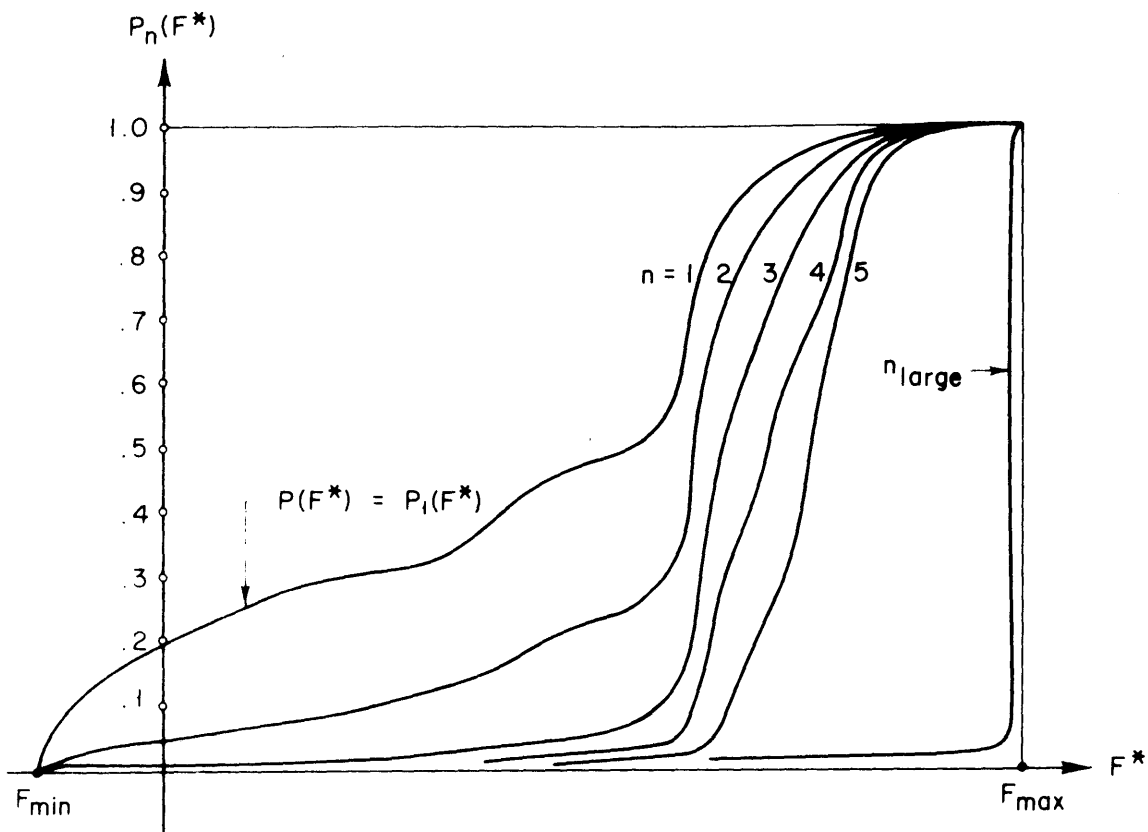


Figure 2-13

A Sketch Showing the Manner in Which $P_n(F^*)$
Varies with n

$P_n(F^*)$ is the probability that after n trials with a pure random scan the largest trial value will have been less than a particular value, F^* . $P_n(F^*) = (P(F^*))^n$.

F^* is quite a bit lower than F_{\max} .

As $n \rightarrow \infty$, the distribution approaches a step distribution;

$P_n(F^*) \rightarrow 0$ for $F^* < F_{\max}$, $P_n(F^*) \rightarrow 1$, for $F^* = F_{\max}$ so that eventual statistical convergence is assured.

Figure 2-14 shows a simple graphical method for visualizing the convergence of the mean value of F^* after n trials. Suppose $P_n(F^*)$ were known. Then the expected value of F^* would be given by the relation $E(F^*) = \int_0^1 F^* dP_n$.

This can be interpreted as a net area on the graph of $P_n(F^*)$ vs. F^* as shown in Figure 2-14(a). Of course, only if $F_{\min} < 0$ will there be a necessity of considering negative areas. Figure 2-14(b) shows how for $n \rightarrow \infty$ the area approaches $F_{\max} \times 1$ or $E(F^*) \rightarrow F_{\max}$.

Figure 2-15 shows two possible distribution functions, one of which will lead to very fast convergence of F^* to F_{\max} , and the other which will lead to very slow convergence.

It is interesting to note that value convergence of a pure random scan is determined completely by the distribution $P(F)$, not by the number of independent variables of the problem or by the complexity of F , i.e., the continuity or lack of it, number of peaks, etc. This has led to the erroneous conclusion in (Brooks²) that convergence with random methods is independent of the dimension of the problem, and hence random methods are preferable for functions of many variables. This statement, while supportable in the case of the Monte Carlo methods, violates common sense

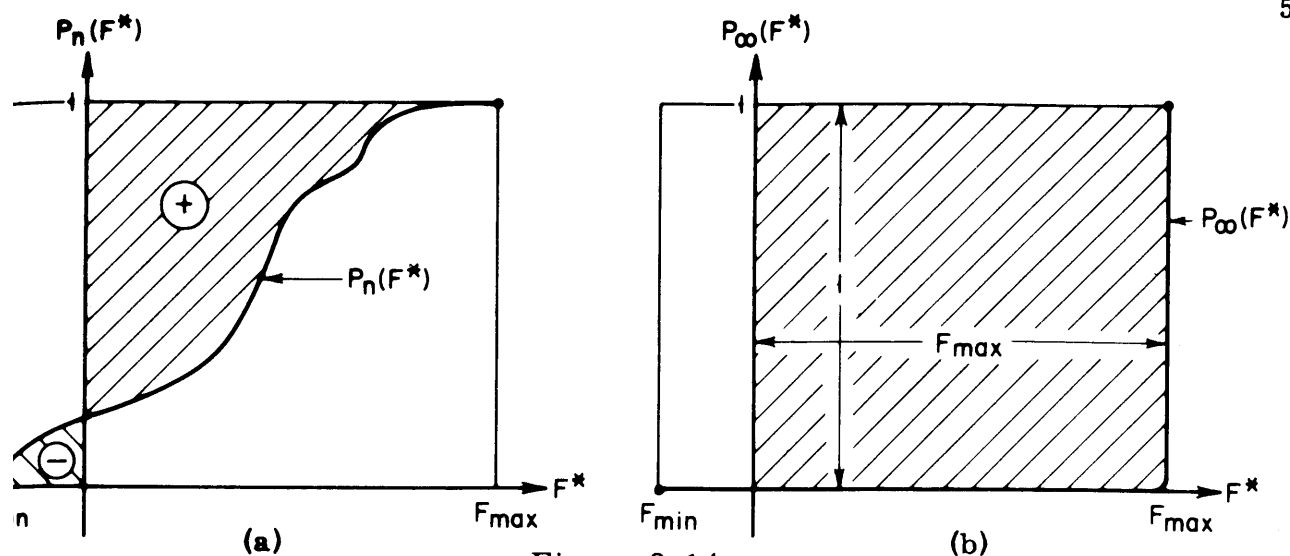


Figure 2-14

The Expected Value of F^* Represented as an Area

In (a), the $E(F^*)$ is represented by the difference between the area marked $+$ and the area marked $-$.

(b) is a demonstration that $E(F^*) \rightarrow F_{\max}$ as $n \rightarrow \infty$.

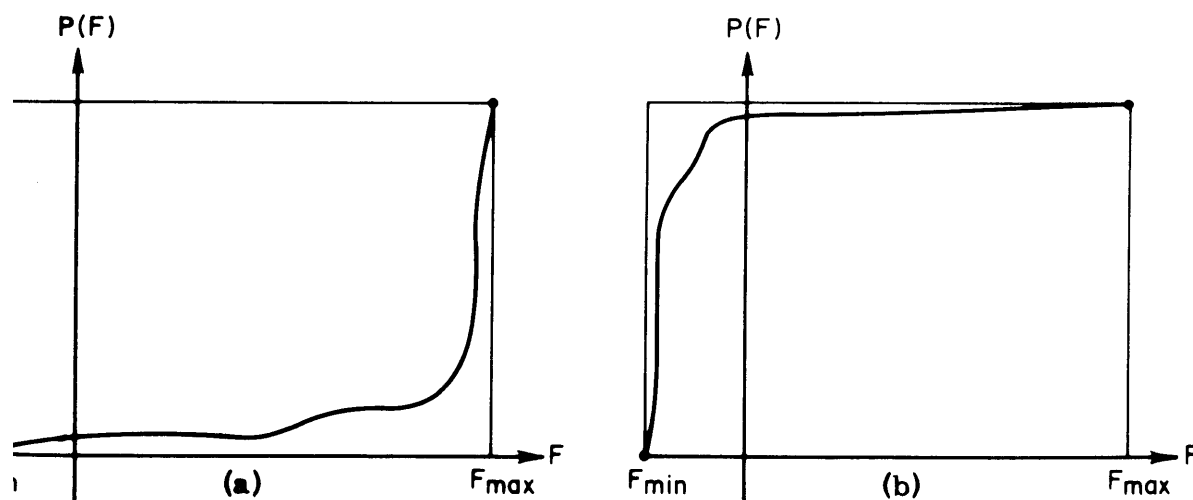


Figure 2-15

Two Possible Distributions of F Values with a Pure Random Scan

The distribution in (a) will result in very fast convergence to the maximum, while the distribution of (b) will result in very slow convergence. Distributions of type (b) can be expected when searching a space of many dimensions.

in the case at hand. In fact, while any given $P(F)$ could have come from an infinity of functions of any number of variables, distributions such as 2-15(a) are likely to come from functions of a small number of variables, and distributions such as 2-15(b) are more often associated with functions of many variables. The strong effect of dimension thus enters into the shape of $P(F)$.

As an example, consider a family of problems, viz. the finding of the maximum of m -dimensional parabolic forms for various positive m . In this case the form and the search space are both assumed normalized so that the search ranges are the same for all m variables, and the form can be expressed as $F(x) = -\sum_{i=1}^m x_i^2$. The scan distribution $s(x)$ is assumed to be rectangular (equiprobable). This example has obvious bearing on the final convergence to the top of any smooth peak in any number of dimensions.

In this case it is possible to get analytic expressions for $P(F)$ and to calculate directly the mean value of F^* for various m as a function of number of trials n . The process should be evident from Figure 2-14. However, these standard operations with probability distributions are not particularly illuminating, and for some m the integrals do not appear to have been evaluated in a closed form. Fortunately, since value and location convergence are directly connected in this example, the same results can be achieved with a very simple argument.

If n_1 points/dimension are tried, a trial will have occurred at an average spacing of $1/n_1$ x full range in each variable. This will require $n = n_1^m$ trial points. Thus in n trials the true location of F_{\max} will have been approached within a distance of the order of $\frac{1}{n^{1/m}}$ x full range. Because of the quadratic nature of F , the average error, $F_{\max} - F^*$ will be of the order of error in location squared. Hence the expected value of the error varies as $\frac{1}{n^{2/m}}$. For example, for a one-dimensional parabola, we have the rapid convergence indicated by the decrement of the error as $1/n^2$. For two dimensions the error goes only as $1/n$, and for three dimensions merely as $1/n^{2/3}$, etc. Parabolic hills in many dimensions are actually analogous to very peaked functions in one dimension. This is brought out further by the nature of the P 's sketched in Figure 2-16, in which it can be clearly seen that there is a transition from the fast convergence case typified by Figure 2-15(a) to the case represented by Figure 2-15(b) as the number of dimensions, m , is increased.

This phenomenon is present in any class of functions, i. e., in higher numbers of dimensions, the search problem rapidly becomes acute. In fact, for a given $P(F)$, the functions must be extremely flat and uninteresting in high-dimensional spaces, while in low-dimensional spaces the functions may be very peaked.

It is instructive to discuss one further question for the pure random scan. Suppose a specific F^* has been found. Further convergence of F^* to F_{\max} can be studied. Clearly, the probability that subsequent values

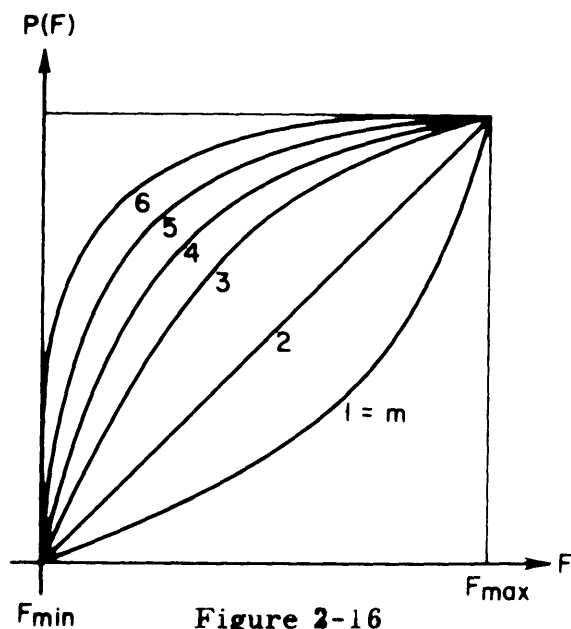


Figure 2-16

A Study of the Shapes of $P(F)$ for Parabolic Peaks in m Dimensions with a Fixed Equiprobable Scan, $s(x)$.

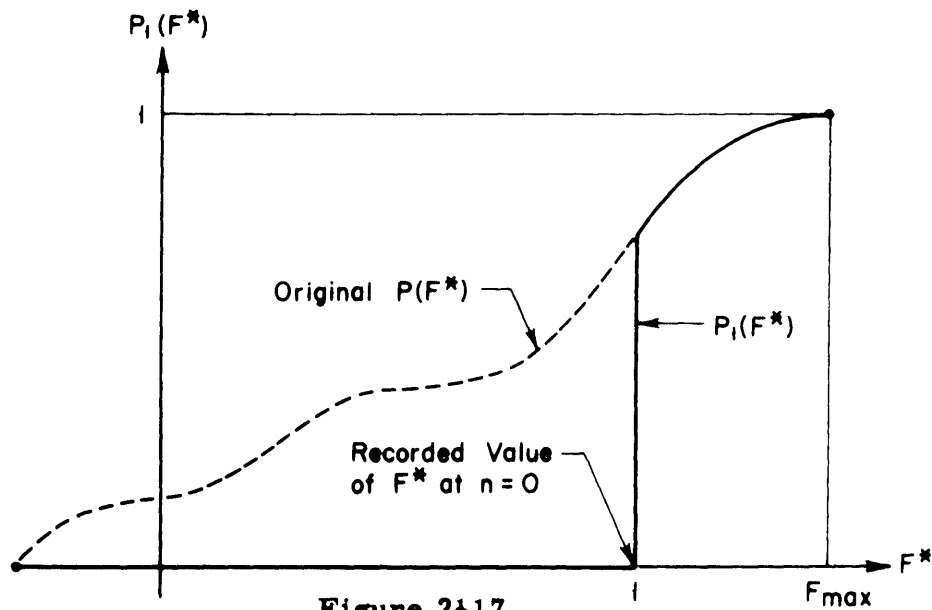


Figure 2-17

The Distribution of Future Values of F^* Given a Particular Recorded Value of F^*

of F^* will be less than the recorded value is identically zero. However, for F^* greater than the recorded value, the probabilities are just the same as those studied previously. Hence $P_1(F^*)$ is just the given $P(F^*)$ but truncated at the recorded value of F^* . See Figure 2-17 which illustrates this case.

Again the distribution for n trials after attaining the recorded value of F^* will be $P_n(F^*) = (P_1(F^*))^n$. Clearly, if the upper tail of the distribution is difficult in the sense that $P(F^*)$ is very nearly unity for F greater than the recorded F^* , then $(P_1)^n$ will not be very different from P_1 until n becomes very large; hence the mean value of F^* will increase very slowly. This is just a more precise statement of the observation that as high values of F are found, it becomes increasingly difficult to find higher ones. When a very close approximation to F_{\max} is desired, it becomes imperative to discover whether some other search scheme can be devised which will produce faster final convergence than the pure random scan. It must be kept in mind at the very beginning that the pure random scan generally produces rapid improvement in the mean F^* . It is only after F^* has approached F_{\max} that the scan begins to be inefficient. How close F^* can come to F_{\max} before the rate of convergence slows down drastically depends on the $P(F)$ in question in the manner demonstrated.

Any method for narrowing the search space is much more difficult to discuss than the pure random scan, for the distribution changes as a function of the trial results. The changed scan distribution has an effect

on the distribution of values of $F(x)$ which can only be discussed if the details of the function are known. If the narrow moving mean scan is used as a prototype of all search narrowing devices, however, most of the advantages and disadvantages which might be expected with any conceivable narrowing scheme can be discussed.

In the first place, to justify a more thorough search in one region of x than another, there must be some reason to believe that a higher value of F than the present stored F^* will be found in the region. The narrow scan located in the vicinity of F^* relies on continuity of the function to suggest that somewhere near F^* there will be a value of F greater than F^* . Clearly this assumption is not valid (1) if the F^* is located on a relative peak, or (2) if the function is not continuous.

While it is quite difficult to discuss the entire convergence of a narrow moving mean scan since too much information about the function is required, the comparison between a pure random scan and a narrow moving scan for a single trial can be discussed. Obviously, in cases in which the average improvement in F^* per trial can be greater with a narrow scan than a pure random scan, it would be sensible to use such a scan at least in conjunction with a wide, pure random scan. The comparison will be illustrated using the area idea of Figure 2-14 to depict the average value of F^* after one trial in various situations.

Figure 2-18(a) shows a narrow scan of the simple hypercubical shape centered over the recorded location of F^* .

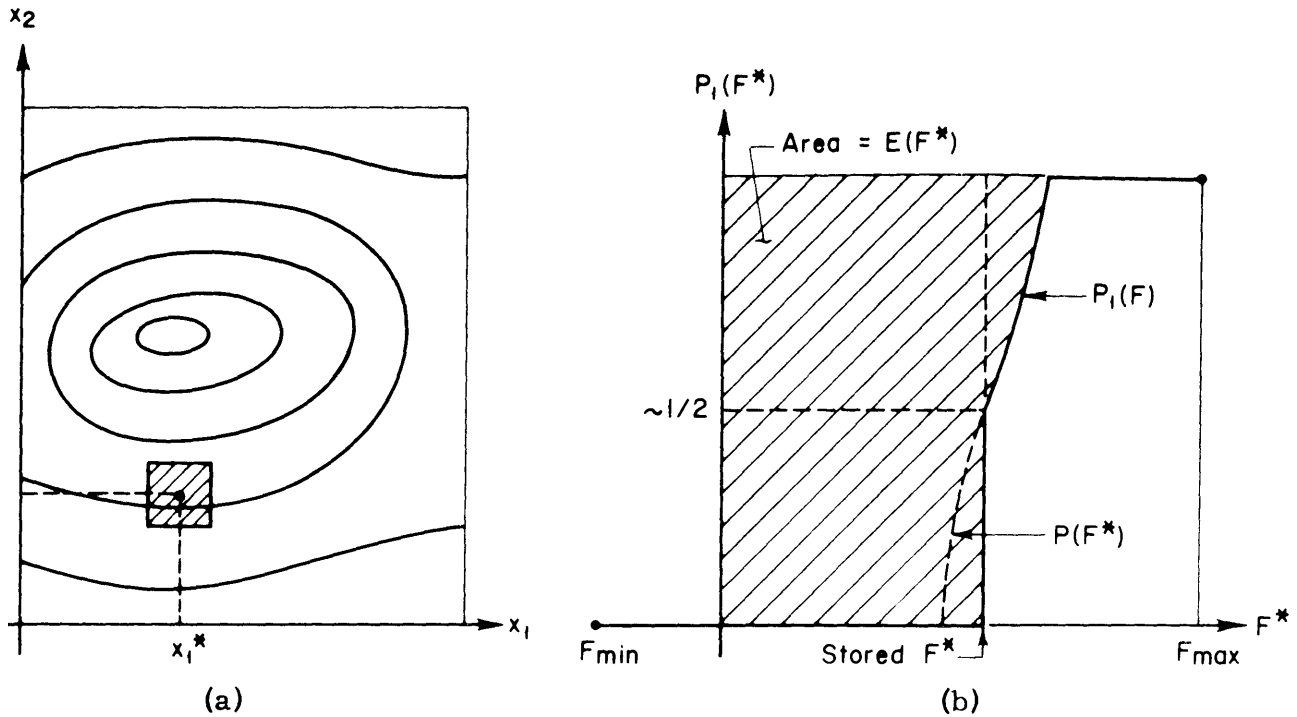


Figure 2-18

Value Convergence with a Narrow Scan

In (a) the narrow scan is shown centered over x^* on a smooth function to be maximized.

In (b), $P_1(F)$ for the narrow scan is plotted. The shaded area represents the average new value of F^* after one trial. The small triangular area in the upper right represents the improvement in the expected value of F^* on one trial. This plot cannot be used for subsequent trials since the scan changes position with each success.

Figure 2-18(b) shows the distribution of trial values using just the narrow scan $P(F)$ and the truncated version of this $P_1(F^*)$ expressing the distribution of F^* after one trial.

One should note that if the scan is narrow enough so that the local region of F is nearly a hyperplane, then the chances are about 50% of finding a value of F less than F^* for the centered scan. As Figure 2-18(b) indicates, a small triangular area above the stored F^* indicates the average improvement in one trial.

Figure 2-19, shows the comparison between a pure random scan and a narrow moving mean scan for three different stored values of F^* . From these graphs it can be clearly seen that the improvement per trial is generally much better for the pure random scan than the narrow scan when F^* is relatively low, and the situation is reversed when F^* is near F_{\max} .

Very good convergence will always occur if the narrow scan is large enough to overlap the true location of F_{\max} but otherwise is as small as possible. However, in practice this is very difficult to achieve, since the requisite information is not usually available. As has been mentioned before, there are many ways for the narrow scan scheme to misbehave. In particular, when the search is narrowed prematurely on the presumption that F^* is near to F_{\max} and hence that probably x^* is near the location of F_{\max} , the scheme may hang up on a relative peak or take a very long time to converge. On a ridge-like function the narrow scan

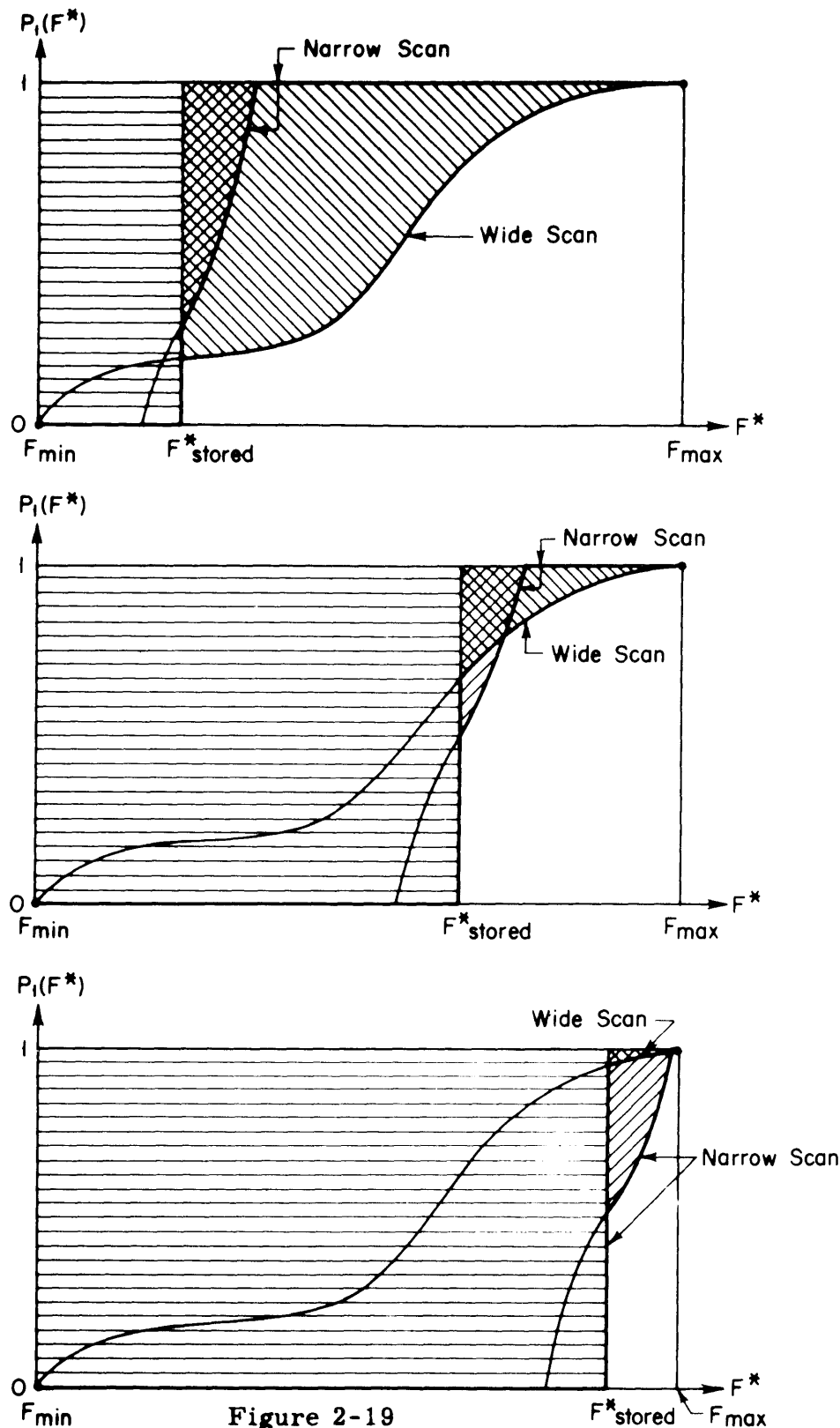


Figure 2-19

Comparison of a Wide, Pure Random Scan with a Narrow Scan
Centered at the Location of F^*_{stored} .

The total shaded area represents the $E(F^*)$ after one trial. The region shaded thus $////$ represents the increase in $E(F^*)$ for the wide scan and the region shaded thus $////$ represents the increase in $E(F^*)$ for the narrow scan. Note that the wide scan is effective for the initial search but the narrow scan is effective for refinement, i. e. when $F^*_{\text{stored}} \rightarrow F_{\max}$.

may get onto the ridge rather quickly by climbing the steep side but be only able to converge very slowly from there on if the slope of the ridge is very gentle along its backbone. In these cases, keeping a wide scan going might be very profitable.

It would seem that the main mistake that has been made in searching functions is to use a refinement scheme too early in the game and to try to force it to search without considering the search problem on its own merits. The real problem is to recognize when a refinement scheme will pay off, and this will be the subject of a later section.

Location Convergence

The finding of a good approximate location, x^* , to the true location in x for which $F(x) = F_{\max}$ is enough different from the problem of estimating F_{\max} that it warrants a separate discussion. In the case of a pure random scan, it is easy to make some statements about the accuracy with which x^* approximates the true location. For instance, given the distribution $s(x)$ and the best trial location, x^* , after n trials, the average spacing of trial points in the vicinity of x^* can be computed. When $s(x)$ is not an equiprobable scan, the density and hence average spacing of points will vary with position in x . The statement then is that the distance between x^* and the true location will be of the order of the trial spacing in the vicinity of x^* or greater. This statement offers a useful bound on the average location accuracy but does not confine it.

The reason, of course, is that in problems with relative peaks, ridge-like functions, or highly discontinuous functions there may be widely separated regions in x that will give the same value of F even as $F \rightarrow F_{\max}$. The reason that the minimax criterion was found to be not very useful in high dimensions was that very difficult functions exist in which the location of F_{\max} is not well tied down even after extensive experimentation. It would be no trouble at all to think of functions for which the average location x^* had virtually nothing to do with the true location of F_{\max} even though F^* might be very close to F_{\max} . An example is shown in one dimension in Figure 2-4(a). Therefore, even for rather simple functions the possibility of a gross error in location cannot always be excluded.

The problem of location convergence is, of course, closely tied up with the problem of the utility of search narrowing schemes, for whenever x^* is converging to the location of F_{\max} , a refinement scheme can easily be invented which will greatly facilitate the convergence. However, such a scheme used in the case of a gross error in x^* may not necessarily increase the convergence of location even if the value convergence is speeded up for a time. A typical misleading result might be to allow a steepest ascent type of procedure to climb to the top of a relative peak. In that case the value would improve, but the location x^* might even diverge from the location of F_{\max} . This sort of behavior is unavoidable sometimes, but the danger should be recognized.

For location convergence, then, there are two sorts of statements which can be made. One, in the case of a function which has at least some continuity aspects, concerns the probable accuracy with which at least a relative peak has been located. The other, which requires a wide scan covering the entire x space to be searched, is a statement about how narrow the absolute peak would have to be in order to have missed it. This last statement is the alternative to the statement about accuracy which gives the order of the accuracy if the true peak has been found. See Figure 2-20.

A search problem may often be usefully divided into two parts -- a true search for the absolute peak, and a refinement of the value and location after the true peak has been located under assumptions about the local character of the function. The refinement phase is relatively noncritical in terms of number of points required, and in the following section reasonable ways to generate information for shifting emphasis from searching to refinement will be suggested.

A Comparison Between a Random and Highly Deterministic Scheme

A random scan method will be compared in this section with a procedure devised by M. Leiter for using a gradient method for finding absolute minima (Leiter). This is quite instructive, since in using any highly deterministic scheme for finding an absolute extreme, many peripheral calculations must be performed, and purely theoretical discussions often neglect these extra operations.

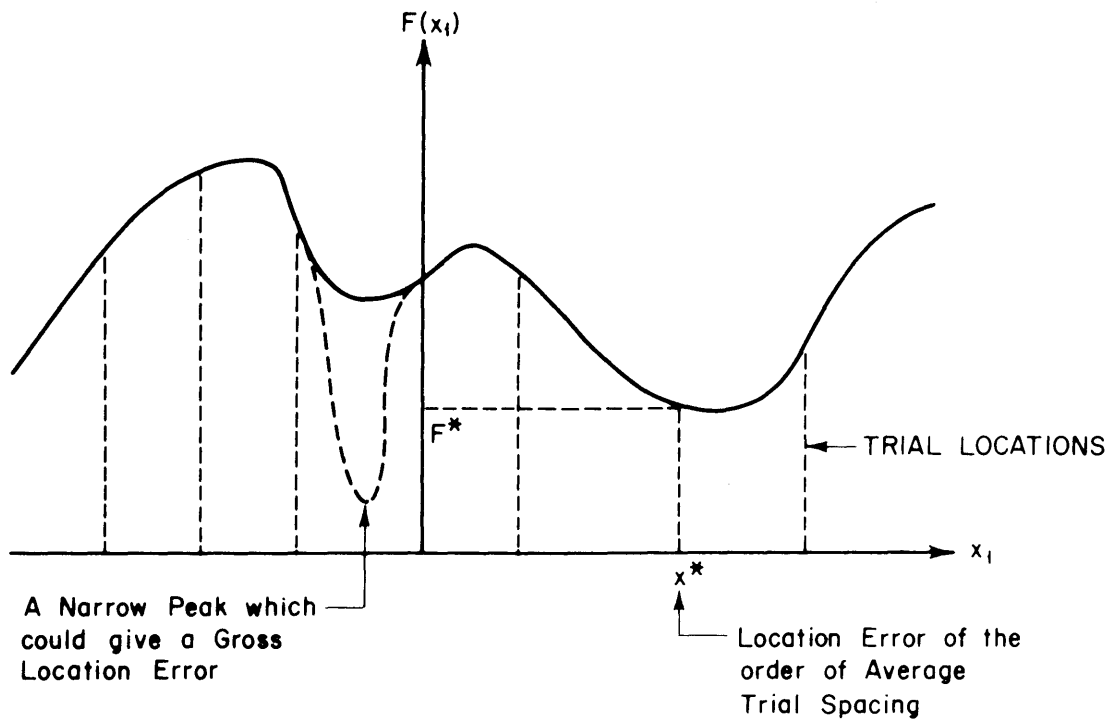


Figure 2-20

Sketch Showing How Narrow Peaks
Can Yield a Gross Location Error

The deterministic scheme to be discussed can be described briefly as follows:

- (1) Choose a random point within the parameter space to be searched (wide range scan).
- (2) Initiate the gradient method.
- (3) Note termination (convergence or overrun).
- (4) Determine nature of singular point (maximum, minimum, or saddle point).
- (5) Draw a zone around the singular point (the volume of parameter space within the zone is to be excluded from the search eventually).
- (6) Test edges of zone to assure that only one singular point is in zone.
- (7) Scan randomly near the zone to see if other singular points are nearby.
- (8) Choose another random point (wide range).
- (9) Initiate gradient until either another singular point is found or the scheme starts to enter a zone.
- (10) Continue until a preset number of random starts have been accomplished.

This is quite a complicated program occupying some 20 pages of SAP machine instructions. One advantage of highly random methods is the dramatic reduction in logic necessary on the machine. Thus, while

Leiter's purpose in devising his program was to show how an adaptive mechanism could be built, the form of the technique requires a complicated general purpose computer for its realization. This would certainly be restrictive in many cases.

A problem tested by Leiter involved finding the real solution to a pair of nonlinear simultaneous equations in two variables;

$$f_1(x_1, x_2) = 0, f_2(x_1, x_2) = 0. \text{ This was done by minimizing the quantity } F = f_1^2(x_1, x_2) + f_2^2(x_1, x_2).$$

Of course, this is not typical of the minimization problems being studied, since the minimum of F is known to be zero, and there may well be more than one solution with exactly the same minimum value.

It turned out that there were twelve solutions (at least in the range $-10 \leq x_1, x_2 \leq 10$), and the computer found them all in 6.4 minutes. The time required to compute F any point in x_1, x_2 was about 3.3 milliseconds.

Much of what is accomplished by Leiter's program can be accomplished in simpler and more efficient ways by means of a proper balance of emphasis between search and refinement techniques. For example, in the test problem, the twelve zeros of F could have been located roughly by scanning the x space and recording all locations for which $F(x) \leq e$ with e an appropriately small number. Even a very rough sampling of the function would enable one to choose e such that only a few points in x clustered around each zero would be recorded.

Since the zeros in the test problem were well separated, a 1% scan would be more than sufficient to roughly locate all the zeros. Such a scan would involve $100 \times 100 = 10,000$ trial evaluations of the function which

would take less than $1/10$ the total running time of Leiter's program. At this point, refinement techniques can be applied to locate any or all of the zeros with great accuracy. This would be the appropriate time to use a gradient method or a succession of narrowing scans. With good approximate locations available, such techniques usually require very little time to achieve high precision.

A random method for refining any one of the zero locations could proceed as follows. Assuming that gross errors have been avoided in the first 1% scan (see Figure 2-20), the center of the small cluster of points surrounding a zero will locate the zero to an accuracy of the order of 1%. Therefore, using the techniques involving the recording of F^* and x^* in a search space somewhat larger than 1% of full range, one can rapidly improve the accuracy. Even as few as 100 trials spent in a 2% space will yield an x^* good to 0.2%. The next step might be to spend 100 more trials in a 0.3% space to reduce the error to about 0.03. Continuing in this manner, every succeeding scan can reduce the error by a factor of $3/20$ at a cost of only 100 trials. Leiter's results are presented to 0.01% accuracy and this can be achieved using less than 300 trials for each zero. Even if all twelve zeros were desired to this accuracy, the 3,600 refinement trials do not take very much time compared with the 10,000 trials for the first wide search. All told, the random method would take considerably less time to accomplish than Leiter's program. The situation would be even more dramatic if an absolute minimum problem were attacked since in the random method only one refinement process would be

required.

Leiter's program seems inefficient for absolute minimum problems for several reasons. In the first place, the randomness enters only in the initial points of gradient searches. Many of these gradient method trajectories lead to the same relative minima and hence do not yield much useful information. Also all the relative minima are found to the same accuracy despite the possibility that a low value of the function may have been found making it logically impossible that the relative minima could be the absolute minima. Finally, a large amount of time is spent in Leiter's program doing auxiliary calculations which are not necessary in the simple random methods.

Although the random scheme could be complicated in various ways (such as making the narrow searches have moving means or noting when the best value occurred at the boundary of a narrow search), it would hardly be conceivable that the process need get as complicated as Leiter's. A very quick count gives about 15 quantities which must be given assumed values at the start of his program. These involve increments for derivative calculations, zone sizes, step sizes, and the like, all of which must be related to the unknown scaling of the problem.

On the other hand, only a few (3 or 4) such quantities need be assumed in the random methods, and only one consideration is really critical; that is, the number of trials on the first scan needed to avoid gross errors. This alone would seem to be an outstanding advantage of the scheme.

The main amount of computing time in the random method is in the first scan. This scan must roughly locate the absolute peak, eliminating the gross errors in location that are possible in too rough a scan. Once this peak is established, the continuity of the function allows rapid refinement.

Obviously, the time required for the first scan is the heavy exponential function of the number of dimensions in the parameter space. On the other hand, during the refinement phase the narrower and narrower subspaces to be searched result in a diminishing number of trial points required for an increase in accuracy. Thus increasing dimension drastically increases the number of first scan points, but during the refinement the number is reduced in just the same way.

This is quite easy to state mathematically. The average location accuracy of x^* for N_1 trials in an m -dimensional space has been found to be $1/N_1^{1/m} \times (\text{full range})$ If gross errors have been avoided.

In the second stage the location is refined by searching a space somewhat larger than $1/N_1^{1/m} \times$ the full range of each parameter, say, $a_2/N_1^{1/m} \times (\text{full range})$. Using N_2 trials, accuracy of $1 \cdot a_2 \times (\text{full range})$ is achieved.

$$\frac{a_2}{(N_1 \cdot N_2)^{1/m}}$$

For high accuracy, "k" successive scans are used. The total number of trials $N = N_1 + N_2 + \dots + N_{k+1}$. The final accuracy of location in the parameter space (assuming sufficient accuracy in distinguishing neighboring points) is of the order of $\frac{1 \times a_2 \times a_3 \times \dots \times a_{k+1}}{(N_1 \times N_2 \times N_3 \dots \times N_{k+1})^{1/m}}$ x(full range).

To simplify the expression one might say all refinement stages with $a_j/N_j^{1/m} = \text{constant} = a_2/N_2^{1/m}$. Then $N = N_1 + k(N_2)$. The accuracy is of the order of $1/N_1^{1/m} (a_2/N_2^{1/m})^k$ times full range.

The swift refinement is indicated by the fact that as the stages progress, the number of trials adds while the accuracy multiplies. The only real problem then is getting the initial accuracy. Of course after the true peak is found, a gradient method could be devised that would be very swift at refinement. However, our analysis indicates that it is hardly necessary.

It is worth noting that if the accuracy in evaluating the function to be optimized is not great, few if any refinement stages will be worthwhile. For example, for many analog computer problems a pure random scan may be all that is needed.

2/2/7 Pattern Recognition Scheme

In previous sections the versatility of random search methods has been demonstrated. In particular, it has been shown that simple changes in the random scan distribution can result in behavior much like that of

the traditional deterministic search methods. The sometimes rapid convergence of a gradient method to a relative peak can be attained with a narrowing, moving mean random method. In fact, final convergence to the top of a peak was found not to be a great problem. Even discontinuities near the peak are not likely to disturb the random scan. However, so far only schemes for using local information to narrow the search have been discussed. These schemes are worthwhile only if the absolute peak has been located. No criterion has been suggested yet for deciding when the absolute peak has been found. It is true that as the scheme finds high values of F , the probability of finding a higher one must drop, and hence the wait between jumps in last best values, F^* , will increase. In the case of several narrow peaks, however, it is clear that this phenomenon might occur even if the scheme were on a relative peak.

Methods will now be proposed for gathering and using global information during the random scan which may help in deciding when to shift emphasis from broad search techniques to refinement techniques. Obviously the general pattern of the function must be discerned. This is the sort of process a human must use to pick out the highest peak of a mountain range without being able to make accurate measurements and comparisons.

While the methods proposed below are evidently only rather crude first attempts, the novel ideas behind them should be noted. Once the fact is faced that many functions require a wide, global search to locate

the region of the absolute extreme before refinement techniques can be fruitfully employed, the possibility exists of condensing and using the information about the general character of the function discovered during a wide search. Clearly, it is not desirable to have to store the location and value of every trial point if this can be avoided. But as much information as possible must be brought to bear on the question of how much wide searching is enough. An explicit discussion of this crucial question seems to be entirely lacking in the literature of maximization problems.

Pattern recognition is an embryonic field which is now receiving much attention. For a general review of current efforts in this line see (Minsky). In many ways this subject treats just the sort of global information that is required to justify a switch from search to refinement methods. Although the field is not well developed, there are clear indications of the sort of operation which must be performed. As before very elaborate schemes will not be discussed, since they probably would work only on a small set of functions anyway. Rather the interest will be in simple methods based on the random scan idea and broadly applicable to all fairly easy problems. The schemes will work well on well scaled and well behaved functions.

Many studies in pattern recognition seem not applicable to general problems. For example, it does not seem reasonable to test whether a function is a distortion of one of a set of prototype functions in the same

same way that a handprinted letter can be tested to see which prototype letter it most resembles.

On the other hand, the techniques in pattern recognition studies involving property listing do seem to be applicable. For example, the property of a function of having either one peak or many would be of interest. In the case of several peaks an indication of their average "width" and "separation" would certainly be important. It must be kept in mind, however, that the use of an incomplete set of properties to draw conclusions and make decisions will always involve risk. Also with only a finite number of trials, mistakes will often be made in estimation of the properties themselves.

So far it seems that the discovery of useful properties has been left to human intuition and that will be the course taken here. It is interesting to note, however, that this process can be automated somewhat if a set of "typical" functions in a particular application could be found. The procedure would be to test various lists of properties and the decision rules based on them to see which combination gave good results most often. In any repetitive situation the computer can learn to make good decisions based on certain estimated properties which have been found effective for describing the sort of functions that arise. In a case such as this the machine would be making a Bayes'-like solution to the decision problem automatically. Of course the system could not try all possible properties, and what properties it finally selected would depend on the test functions. Thus

the scheme would not be much good for problems dissimilar to the test problem.

Two general techniques are discussed below which seem compatible with random scan techniques.

One source of information about the function in question is the distribution of F values generated by a temporarily fixed random scan in x . A way to estimate this distribution is to bracket the possible values of F and to count how many trial values of F fall into each bracket during the wide range search. The histogram thus generated approximates $p(F) = \frac{dP(F)}{dF}$. Using the histogram, properties of the function such as the mean value and the range (relative to the particular $s(x)$ used) can be estimated. In addition, the existence of a lucky strike in which a value of x is chosen on a very high peak is made quite evident. Figure 2-21 illustrates typical histograms. The histogram requires very little storage but contains much information about the function which can be incorporated into decision rules for the search. For example, when a high sharp peak is found, probably refinement is worthwhile. On the other hand, the histogram may indicate that the function is so flat that there is little to be gained in finding the optimum.

The histogram is useful in that it preserves the essentially one dimensional character of a pure random search for the value of F_{\max} . It does not say much directly about the x behavior. Any histogram could come from a great variety of functions even including those densely discontinuous.

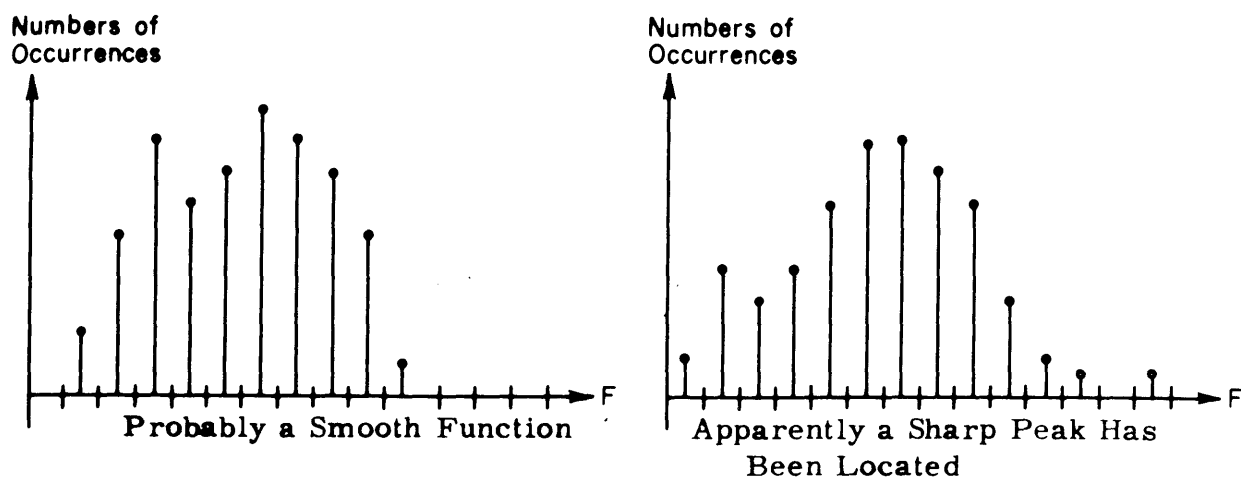


Figure 2-21

Two Histograms Used to Condense and Present the Results of the Trials

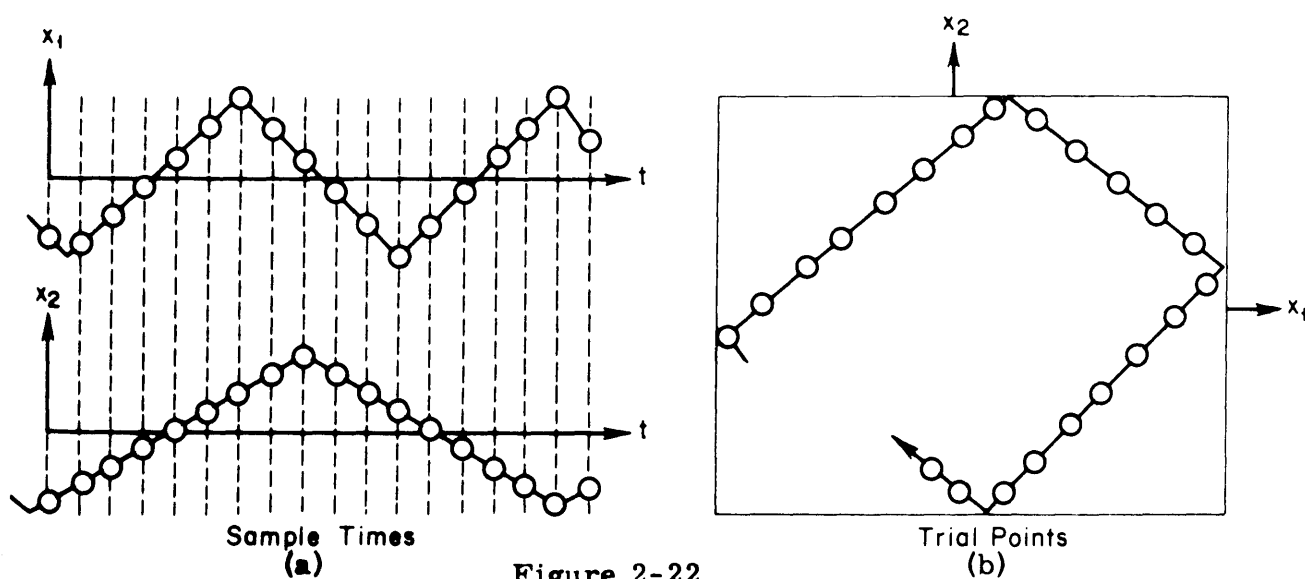


Figure 2-22

Generation of Trial Locations Which Are Correlated in x

In (a) the trial values of x_1 and x_2 are generated by sampling two desynchronized triangular waves. The results are shown in (b). The same principle can be used when the trial locations are arrived at by sampling random waves. All that is sought is that successive trial points shall be close to each other in x .

Information about the x behavior can come from a correlation of x with F . In order to avoid very clumsy data processing procedures involving the storage of x and F values, a simple trick can be used. If the successive x trials are heavily correlated, then the successive F 's can be tested for correlation. Evidently, for small steps in x on a smooth function the F 's should be highly correlated. On the other hand, if the function is badly discontinuous or if the x steps are a large part of peak and separation width, the correlation between successive F 's will be less. Obviously then a sort of global average of the wave length of the function can be obtained by noting the distance at which correlation between F 's breaks down. This will certainly give a lower bound on the number of search points needed. Only if the absolute peak is unusually narrow compared to the others scanned will the system be far in error.

The successive x values can be correlated very easily if a sampled wave method is used to generate the trial points. The sampling must simply be fast relative to the wave frequencies. This procedure is illustrated in Figure 2-22.

In the case of Figure 2-22 the correlation is polarized around only two directions of travel in x (in an m -dimensional space it would be 2^{m-1} directions). A random walk with reflecting boundaries would be somewhat preferable if it could be mechanized easily. The question of which step size or mean step size to use is again not really answerable. In general the steps should be small so that correlation on one step interval will show continuity of the function, but the steps must not be so small that not much

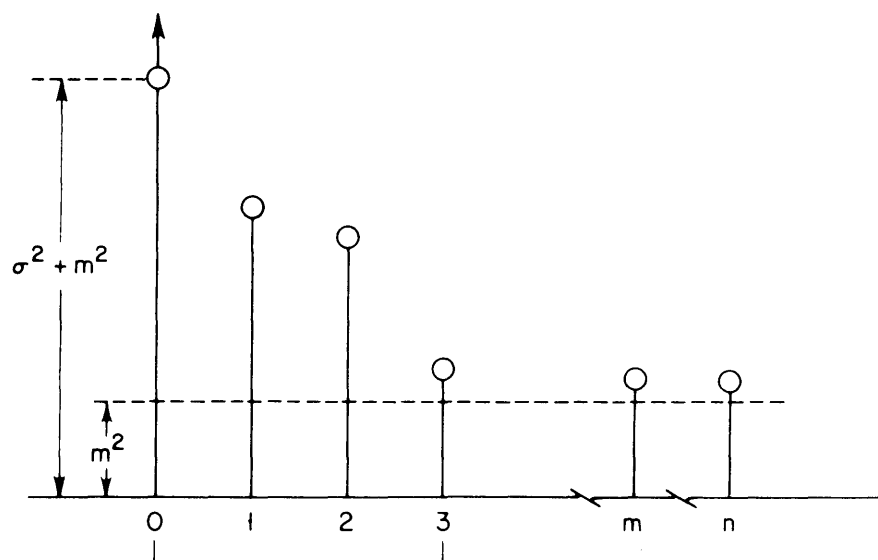
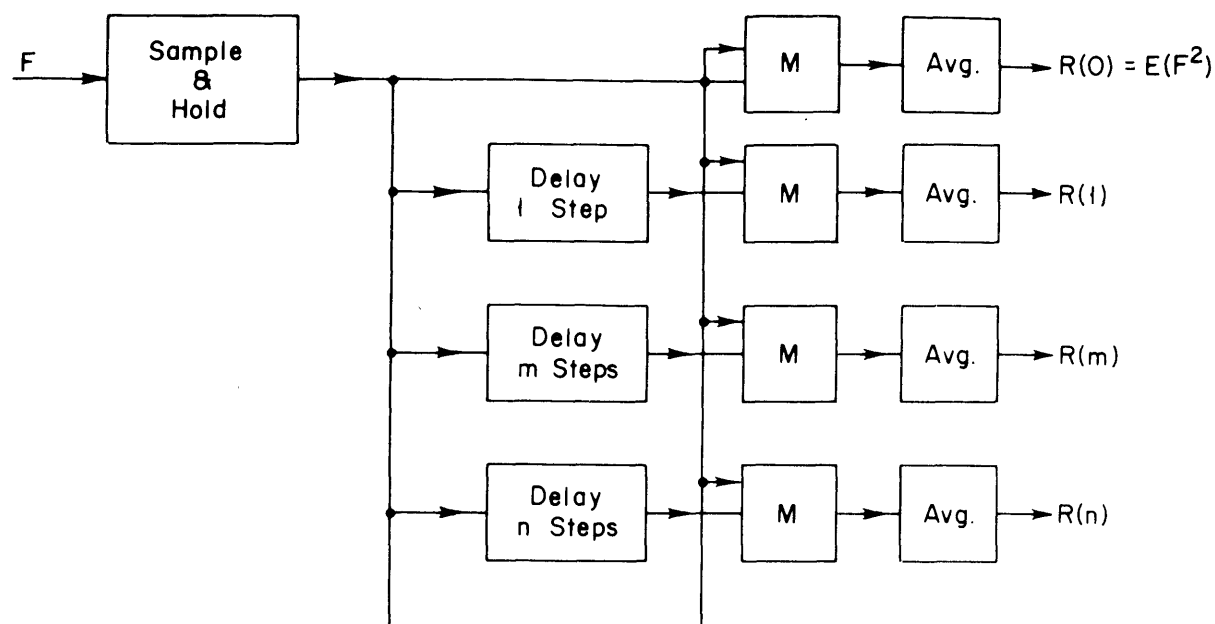
of the function is explored during the duration of the wide scan. The correlations will confirm or deny the wisdom of the first try at choosing the proper step size.

Correlating the F values can be done at the same time as the trials are proceeding by using analog averaging. All that is needed is some kind of a sampler-holder to convert the successive trial F's to a staircase function, a synchronous delay, and a few multipliers and averagers. See Figure 2-23.

Several variations on these correlation methods and uses for the data suggest themselves, including the modifications necessary for a time-variable case. With experience on a class of problems, it is entirely conceivable that one could automate the entire process of using the global information condensed in a histogram or correlogram. For design problems the averages and correlations can be considered as data reduction methods, and a human can make the few grand strategy decisions required during a search.

2/3 TIME VARIABLE FUNCTIONS

So far random scanning methods have been discussed in detail only for the case in which the optimization problem did not change with time. This would often be the case in a design situation and would be approximately the case whenever the function in question changed very little in the time required for the search.



Upper bound on average wavelength and hence a lower bound on the number of trials before refinement

Figure 2-23

Correlation of F Values - Block Diagram and Typical Results

It is instructive, however, to study the time variable case with some thoroughness, since many possible applications of the technique are in this category. The minimum computation speed required by a given time variable problem can be estimated as well as something about the errors in the results of the search due to various causes.

Many results in this section will be qualitative in nature. While some results can be made quantitative in fairly evident ways if certain information about the situation under study is known or assumed, some relationships seem to be so complex that experimentation on the system seems to be the most reasonable course.

With a time variable function there seems to be no difficulty in imagining an instantaneous extreme value at a definite location in the space of the variables, but there is an essential difficulty in computing this point. The difficulty is that trial points computed in successive time periods are not strictly comparable*. Since the value of the function at a particular x will be different at different times, unless all comparisons of function values could be carried out at a single instant in time, the comparability property of 1/1/1 has been lost. The simple idea of remembering the best trial value of F no matter when it occurred is evidently hopelessly conservative.

* The not very intriguing possibility of reducing the computing time by having a number of computers operating simultaneously is not considered.

In order to keep up with the changes in the function, a fairly radical policy must be adopted to insure that fresh experimentation is given precedence over older results. The solution can no longer be refined arbitrarily as in the static case. The general idea will be to minimize the error given the function and how it changes in time and a certain computation speed.

2/3/1 Complexity, Strategy, and Achievement

Just as in the fixed function case several orders of complexity of the search problems must be distinguished. As before a function of a large number of variables is much more difficult than a similar function of few variables. Also discontinuous functions are more difficult than continuous ones.

Time variability is another factor which enters into the basic difficulty or complexity of a problem in an essential way. The easiest or least complex problem occurs when the true peak, continuous in space, moves in a continuous way with time with the restriction that it never sinks to the level of a relative peak. The most complex problem occurs when a function highly discontinuous in space changes discontinuously in time.

As in the static case there are two distinct but related errors that must be considered in the computation of the absolute extreme of a function, i.e., the error in the approximation to the extreme value of the function, F_{\max} , and the error in its location in the independent variable space, x . In time variable problems, the location of F_{\max} is probably often the

most important result, since this result might be used to set the parameters of a physical system for presumably optimal behavior.

In the time variable case one cannot simply discuss value and location convergence, but rather must consider the achievement of a search scheme in maintaining $F^*(t)$ as a good approximation to F_{\max} over a period of time, and the strategy $x^*(t)$ which should always be a good approximation to the true location of F_{\max} .

One way to describe the difficulty of the computational problem is to consider the sensitivity of the problem to these two types of error. As noted previously, a function with large discontinuous jumps in the x vicinity of the extreme is very sensitive to errors in location, and hence the location must be accurately known. In the same way if the time behavior is discontinuous, in a small increment in time the stored solution may suddenly be invalidated. This sensitivity in time implies that a new solution must be available very quickly after a change.

On the other hand, errors in location of a smooth peak need not cause large errors in the approximation to F_{\max} . Such a peak does not represent a sensitive situation except where the peak is very narrow, in which case the peak can be mathematically smooth only in such a small range that for finite accuracy purposes it appears discontinuous. Similarly, a continuous deformation of a continuous surface in time is generally a nonsensitive phenomenon. In this connection it is important to note that if a past location of the absolute peak is stored under continuous

deformation in time, the error between the real F_{\max} and the F^* at the remembered location will change continuously. However, if the absolute peak is superceded by one of the relative peaks, the location of the true peak will change discontinuously. An example of this was shown in Figure 1-8. In this aspect at least continuity in time is less helpful than continuity in space with regard to the location error.

Where there are many peaks with relative extremes near F_{\max} in value, the scheme will be location sensitive. While it may not be a serious situation as far as $F_{\max} - F^*$ is concerned since all peaks may be quite high, the location stored as x^* will not steady down very rapidly. This might have a bad effect if physical parameters are to be set at the computed optimum point, since the physical parameters will spend too much time in transition and too little time at even a relative optimum. Averaging the $x^*(t)$ position would be very bad in this case, although such averaging would be logical if x^* were moving about on the absolute peak due to noise effects.

Knowing now the easy problems from the hard ones, we can set about modifying our scan technique to handle the time variable problem. In many cases the random scan is more versatile than most methods. For example, the moving mean narrow scan, which is superficially like steepest ascent methods, may work quite well on a continuously changing discontinuous (in space) function while any gradient method would behave terribly in this situation.

2/3/2 Modification of the Random Techniques for Time Variable Functions

It is obvious in a time varying situation that some minimum trial rate is required to obtain significant results. This lower bound on the trial rate is fairly easy to estimate under a variety of situations. For instance, if the problem involves a continuous peak moving continuously in time and it is desired to track the location within some given per cent of the full range, enough points must be tried to locate the peak within the specified error in a short enough time interval such that the peak does not move much more than the allowable error while the search is going on. This will set the trial rate, R , if the scan distribution is known and if a rough measure of the speed of the peak can be made.

Keeping in mind that good scaling is assumed so that percentage errors in location can be roughly the same in all directions and assuming a maximum peak velocity $v_{\max}^2 = \dot{x}_1^2 + \dot{x}_2^2 + \dots + \dot{x}_n^2$ and a full range x_{FR} for each parameter, we can calculate the minimum trial rate F for an equiprobable pure random scan as follows.

If F can be considered fixed over a short time interval Δt and if gross errors can be avoided, the order of the percentage error in location will be given as $\frac{\Delta x}{x_{FR}} = \frac{1}{N^{1/m}} = \frac{1}{(R\Delta t)^{1/m}}$ by the formula for a static function. During Δt the peak will have moved at the most an amount $x = v_{\max} \Delta t$ so that while increasing Δt would seem from the first formula to decrease the error, the effect in the second formula is to increase the error.

Given a Δx , the necessary trial rate may be found by allowing only

$$t = \Delta x / v_{\max} \text{ for the search time. The result is } R_{\min} = \left(\frac{\Delta x}{x_{FR}} \right)^{-m} \frac{v_{\max}}{\Delta x}.$$

This trial rate R_{\min} is a minimum because a higher rate may be required to locate the absolute peak and thus avoid gross errors.

All of the assumptions implicit and explicit are necessary if this minimum trial rate is to be approached in practice. It should be thought of merely as a lower bound on R when a pure random scan is used. It does indicate, however, the manner in which the range x_{FR} , the peak velocity v_{\max} , and the number of dimensions m combine to affect R .

The above rough calculation of R_{\min} and other similar ones are based on the idea of accomplishing a number of separate searches, each to be completed in a specified time interval. It is more rational to have the computation proceed continuously. The main difficulty is that the technique for stationary functions of remembering the highest value attained on any trial must be modified in a time varying situation to forget any trial in the remote past.

There is only one situation in which the stationary function technique will work on a time variable function without modification. This is the situation in which F_{\max} is always rising, for in this case the technique will self-correct at least. In the case of a falling maximum on the other hand, a nearly maximum trial would be stored until the F_{\max} should rise again no matter what might have happened in the meantime.

Perhaps the simplest modification that can be made in the random methods that will make them useful for time varying problems is to allow the memory device to forget the last best value of F if it occurred in the remote past. In searching for a maximum then, one way is to make the stored values of F^* drift downward so that eventually the system will always transfer to another point in x which yields a trial value of F higher than the drifted stored value. The drift in this case must be monotonic downward to eliminate the possibility of the scheme being stalled indefinitely. There is the possibility that if the absolute peak falls faster than the memory down-drift, the scheme can be stalled until the drift catches up with the peak. For good peak tracking, the drift rate must be somewhat higher than the maximum downward velocity of the absolute peak. However, in the cases in which drift is not needed, viz. stationary and rising peaks, the effect of drift is deleterious. Another point which can be studied involves the shape of the drift versus time curve. On analog equipment, for example, drifts of linear and exponential form can usually be obtained very easily, but it is evident that given the function, its time behavior and a clearly stated goal, an optimization problem can be stated for finding the optimum drift shape. It would seem that experiment would settle this question satisfactorily in a rather short time in particular cases.

To illustrate the effects of time variability and the necessary modifications to the random scan techniques, the case of a continuous function with continuous changes in time, will be considered.

It is useful to distinguish three types of location errors in the searching process:

(1) Gross errors which appear at the beginning of the search or when the scheme has been stalled on a relative peak for one reason or another.

(2) Dispersion errors which occur as the scheme is trying to converge to the absolute peak, and

(3) Lag errors in which the scheme is left behind a moving peak due to finite trial rates. Actually, only (3) is a phenomenon peculiar to time varying systems, although the time dependency affects the occurrence of (1) and (2).

The gross errors which can occur whenever an absolute peak sinks below the level of a previously relative peak have already been mentioned. Whenever this possibility is not excluded from the problem, the wide range search must be retained. This is quite a severe restriction, since as the studies of the fixed function have shown, wide searches are likely to be very expensive in number of trials required or in the time variable case in the trial rate required. Once again the monomodal function is seen to be a peculiarly simple case requiring mainly refinement techniques rather than true searching.

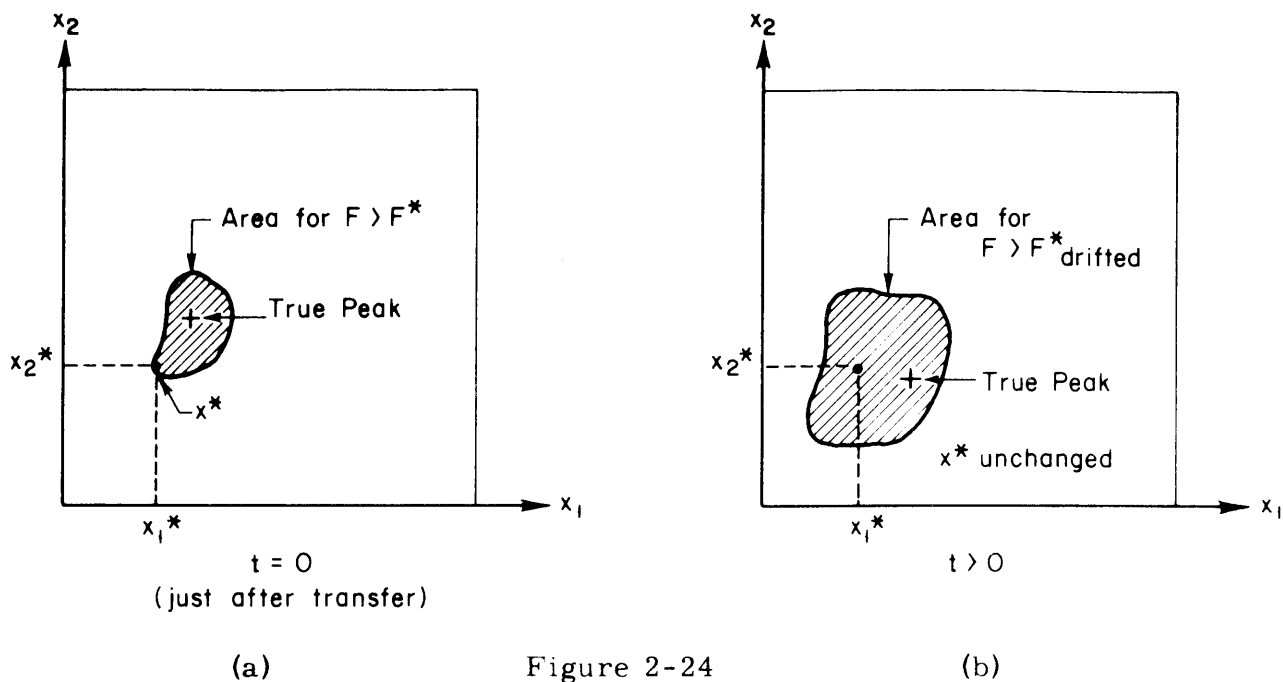
By dispersion errors is meant those location errors which occur as a peak is being approached. For smooth peaks these errors are distributed around the peak and diminish in time for the fixed function case. For a semi-continuous peak the dispersion pattern may not surround the peak; the maxi-

mum may be on the edge of the pattern. For very discontinuous functions or for the case in which a relative peak is almost as tall as the absolute peak, the accuracy of computation may never be sufficient to define either a dispersion or a lag error.

If a high enough trial rate can be assumed so that quite good peak following can be expected, then dispersion can be studied in a semi-quantitative way. If at $t = 0$, x^* is located somewhere on the absolute peak, eliminating for the present the possibility of transferring to a relative peak, then the area in the x space for which a transfer of x^* is possible can be imagined. See Figure 2-24.

In Figure 2-24(a) a transfer in x^* has just been made, and hence if a trial x_1, x_2 falls in some area for which F is greater than F^* , another transfer of x^* will occur. Assuming that the probability distribution is approximately rectangular at least over the transfer area, then all points within the area are equally likely to be tried so that the succeeding error is certainly of the order of the linear dimensions of the area. Now if down drift in F^* is allowed, the transfer area becomes larger; hence the dispersion error for succeeding times becomes larger. See Figure 2-24(b). If the drift continues too far before a new transfer of x^* , areas corresponding to relative peaks and hence gross errors in location may appear.

It is evident too that as time goes on the true peak is rising or falling, and the dispersion area is affected by this in addition to the down drift of the F^* memory. If the peak falls too fast the transfer area disappears altogether and the search is stalled.



The Effect of Down Drift on the Size of the Transfer Area and Dispersion Error

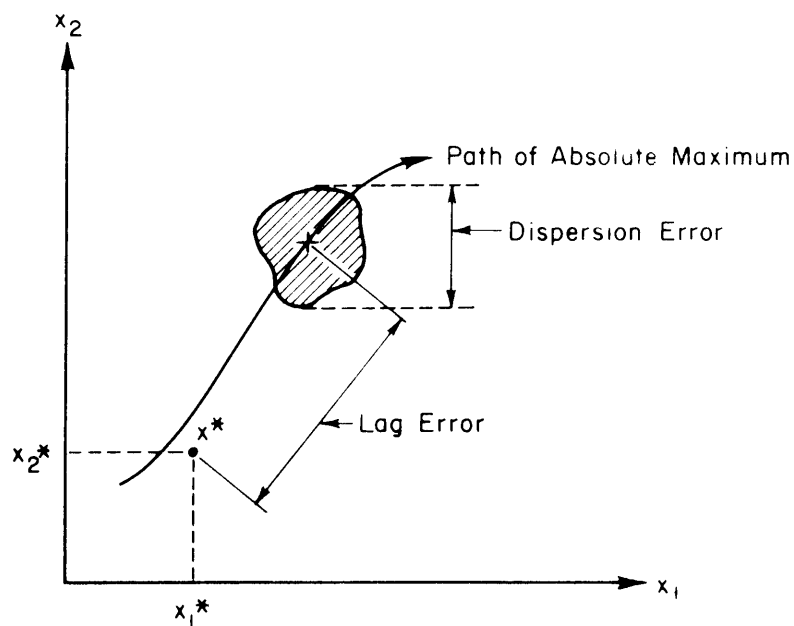


Figure 2-25

Two Types of Location Error in Time Variable Problems

The last form of error occurs when a transfer fails to occur for some time and the peak moves away from the stored point. See Figure 2-25. Evidently, this lag error can be very serious if a transfer fails to occur for some time. This is just the case when the absolute peak has been approached quite closely, since then the transfer area becomes very small. The probability of a transfer in a given time is proportional to the trial rate R and is affected by the shape of the trial distribution and the size of the transfer area. If the distribution is flat over the area, the probability of transfer is proportional to the area or generalized volume in m -space.

Since the problem of transferring or not is a two-state problem, the length of time on the average it will take for a transfer can be calculated if the probability of a transfer, p is known.

The probability of the first transfer on the n^{th} trial is the probability of $n-1$ failures and a final success.

If $q = 1-p$ is the probability of a failure to transfer, then the probability of a transfer for the first time on the n^{th} try, p_n , is $p_n = q^{n-1} p$.

The mean time to transfer \bar{n} shown in geometrical interpretation in

Figure 2-6 is $\bar{n} = \sum_{n=1}^{\infty} n p_n(n) = \sum_{n=1}^{\infty} n p q^{n-1} = p(1 + 2q + 3q^2 + \dots)$.

This series converges for $|q| < 1$ to $\bar{n} = \frac{p}{(1-q)^2} = \frac{1}{p}$. Thus if the peak has a velocity $v^2 = \dot{x}_1^2 + \dot{x}_2^2 + \dots + \dot{x}_n^2$ the average lag error if p should remain constant is roughly of magnitude $v \Delta t = v \frac{\bar{n}}{R} = \frac{v}{pR}$.

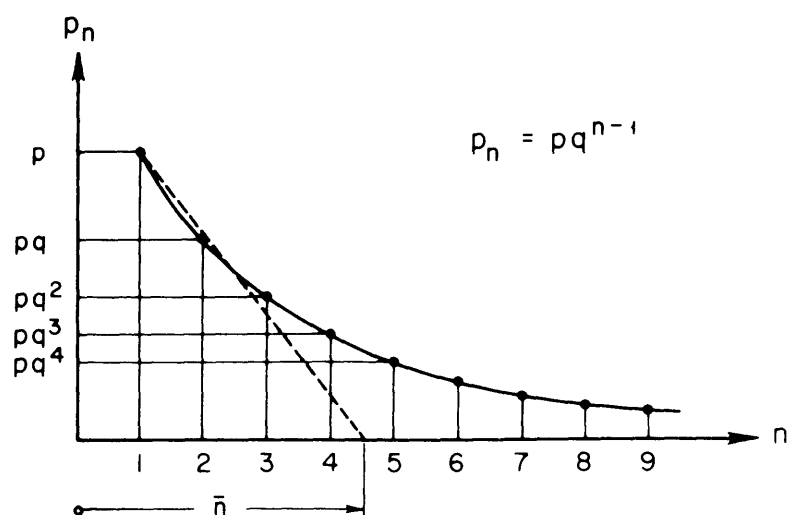


Figure 2-26

The Probability of Finding a Value of F Greater than F^* for the First Time on the n th Trial

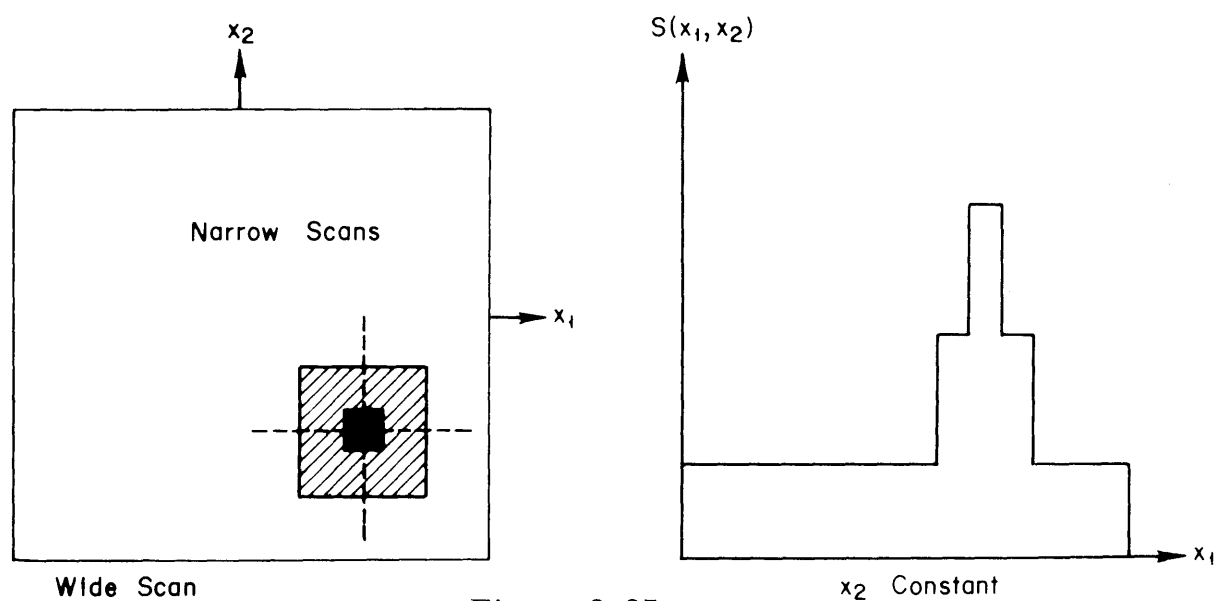


Figure 2-27

A Wide Scan Combined with Two Refinement Scans Useful for Time Variable Search Problems

The drift of the F^* memory serves to reduce the lag error by increasing p . This is done by lowering the stored F^* which through the function $F(x)$ increases the transfer area, and hence the probability of transfer p . The increasing of the transfer area does, however, increase the dispersion error. Obviously the best that can be done is to keep these two types of error about equal in magnitude. Given the function this determines the optimum down drift of the memory. Although we will rarely have enough information to determine the optimum down drift, it is important to note the trading between lag error and dispersion error. One must keep in mind always that the concept of lag error is important only when gross errors have been avoided.

So far the necessity of the wide, expensive search has been discussed, but the refinement ideas are also still valid. Just after a transfer in a system which can follow a peak fairly well, x^* can be expected to be near the true peak very often. Hence a local search of particular intensity is indicated. If narrow searches are superimposed on the wide search area, it may be possible to refine the approximate location of the peak very fast. This technique should be able to much increase p in the above formulas without stealing much from the wide search. In fact, this technique can be used alone in a monomodal problem and is very efficient as long as the peak doesn't slip out from under the narrow scans. The idea of successive refinement can even be used in a continuous form.

See Figure 2-27.

The refinement stages are rather inexpensive in terms of trial rate. In this way very high density searching can be achieved at little cost in small regions, and under favorable conditions very good peak tracking can be accomplished.

The implication is that problem solvability is almost a step function of trial rate available. If R is below a minimum very little of value can be done. When R exceeds some value the problem is solvable, and if there is even a rough form of continuity the scheme can do very well most of the time. Somewhat surprisingly, these random methods which approach maximum efficiency on general problems still have quite trivial programming.

CHAPTER 3

OPTIMUM AND ADAPTIVE SYSTEMS

There are two reasons for discussing the topic of this chapter. The first is that the application of the search techniques discussed in Chapter 2 will generally involve searching for an optimum system or constructing a self-optimizing or adaptive system. The second reason is that the search schemes themselves have many characteristics in common with adaptive systems. In particular, some of the fundamental difficulties in the construction of an optimum search method have analogies in the difficulties which arise in an attempt to realize an optimum or optimizing system. In Chapter 2 reasonable methods for searching a function were discussed; in this chapter reasonable ways for constructing adaptive systems based on search techniques will be discussed.

3/1 SUGGESTED OPTIMUM SYSTEMS

Before discussing adaptive or self-optimizing systems, it would seem reasonable to consider the so-called optimum systems which appear in the literature. One might naively expect that a self-optimizing system would approach the characteristics of one of these optimum systems after a while, but this seems rarely to be the case. In fact, if an optimum system could be designed, there would hardly need to be a theory of self-optimizing systems.

The first characteristic to note about any discussion of optimum systems is that there is a distinct framework and range of possibilities out of which the optimum system is chosen. For instance, the well known optimum linear filter theory of Wiener (Wiener²) does not necessarily always produce a very good filter. The possibility is not precluded that in some situations a very simple nonlinear filter might be found that would perform better than the optimum filter in the Wiener sense. It must be realized that the least bad of a series of bad alternatives could be called an optimum. Thus the use of the word optimum, which seems to imply an absolute standard, can be quite misleading.

Another vexing problem about optimum systems involves the question of physical realizability. Since no realizable device can have a perfect pre-knowledge of the future, an optimum system theory which relies on such knowledge may not be very useful. A vital step in the Wiener filter theory is the introduction of a physical realizability condition, but in a larger sense the optimum Wiener filter is still unrealizable. This is because the optimum filter is a function of the statistical characteristics of the input signals. Thus an optimum filter is constructable only if the statistics of the future are predictable. Clearly, it is easier to predict statistics of a time function than the function itself, but philosophically the optimum filter is actually unrealizable.

Another example is the maximum effort type of optimum system theory (Bellman², Pontryagin, Rozanov). In this case the forcing

variables acting on a system are merely bounded, and an optimum time history of these forces is sought when the system is asked to perform some operation such as returning to an equilibrium point. In this sense the optimum behavior is well defined. However, in a real control situation such an optimum system may not behave well at all. The difficulty can be illustrated by a simple example. Suppose a company wires its Boston representative to get out to Los Angeles as fast as possible, with the understanding that he use commercial carriers. The optimum salesman will certainly get on the jet plane with the earliest scheduled arrival in Los Angeles, since this is his maximum effort solution. It is obvious that the plane may never even get to Los Angeles due, for example, to a crash. In that case, a non-optimum salesman who took a train might have performed better in the end. Of even more interest would be the case of a clever salesman who also took the train because he remembered that the front office was always changing its mind. Then if a counter order came to go to Pittsburgh instead of Los Angeles, it is possible that the maximum effort salesman would be stranded in Saint Louis, while the sluggish but clever salesman would be on the spot in Pittsburgh.

Another problem in defining a truly optimum system, which is also tied up with future prediction, is that present actions will have a somewhat unpredictable effect in the future. Paradoxically, in the case of final value control in which all operations are directed toward a specific

future state, the problem, at least philosophically, does not seem as bad as in normal control problems. In the salesman's final value problem of getting to California, there is only a finite amount of the future to worry about. The salesman could think of all possible ways to achieve his goal and pick what he imagined was the best.

The problem is quite different for any control criterion which extends into the indefinite future. Most businesses, for example, wish to maximize profit not just for a fixed time but presumably forever. Obviously, what is good for the short run may be disastrous in the longer run. To imagine finding a policy which will assure goals for the infinite future is surely folly. Yet simply optimizing for the short run will not do. The novelty manufacturer who maximizes profit on a "fad" one week may be out of business the next as the populace tires of his device. Also to get from Boston to Los Angeles in the fastest manner, it will not do to simply maximize westward velocity at every instant. Such a policy would require an immediate departure on foot in a westerly direction even if it were known that in a few minutes a plane was scheduled to leave for the west coast. This is obviously the beginning of a very poor strategy. These examples demonstrate that the "principle of optimality" of Bellman's Dynamic Programming does not hold for all processes. If it did, what was optimum for the short run would be the beginning of an optimum procedure for the long run.

The point to be made here is that in the definition of any optimum system a number of assumptions must be involved, perhaps the least appreciated of which are concerned with future predictions. An actual test of an optimum system might prove very disappointing if the time history of the test was not of the type assumed or if the optimum system was compared to any system out of its class which happened to be appropriate to the particular test.

3/2 ADAPTIVE SYSTEMS

Adaptive systems may be usefully regarded as systems which do not make long-term predictions about the future, but rather readjust themselves in response to measured changes in the environment or their own characteristics which are predicted to persist at least for a short time. The main characteristic of adaptive systems, then, is a sort of flexibility, an ability to vary operating behavior in response to unpredictable variations in any quantity which affects the performance of the system. If the adaptive system is truly self-optimizing, then if the situation remains stationary long enough, the system will approach the optimum system for the existing conditions and relative to the totality of adjustments that the adaptive system is capable of making.

It seems clear that the problem of adaptive systems can be stated as a time variable search problem, although not many authors of papers in this field have approached the problem from this viewpoint. We shall take the view here that the adaptive control problem is a search problem,

and shall suggest reasonable ways to construct an adaptive system which utilizes the general flexibility of the random search procedures. In this way the technique can be suited to particular situations, and hopefully good results can be attained in many cases. However, since an adaptive system attempts to become an optimum system at least for the short run, the difficulties discussed above for optimum systems cannot be completely avoided. In addition, the adaptive system will take some time to achieve adaptation, and hence the system cannot behave well if it tries to adapt to effects which change too rapidly.

The main difficulty in the design of an optimum or adaptive system is very like the difficulty which usually prevents a minimax strategy in the problem of the search for a maximum. In the search problem it was seen that for any reasonably large class of multi-dimensional functions a finite number of observations cannot be guaranteed to restrict the possible locations of the extreme. In the present problem of adaptive systems, the knowledge of the past history of everything connected with the problem cannot guarantee very much about the future, but the system is forced to make decisions in the present which will affect future performance. What is true of course is that at least the near future can often be predicted quite well, as well as some aspects of the more distant future. Hence an adaptive system may be able to keep quite near to the optimum relative to a short run at least, and with luck this may be quite a successful policy. The point is that it is extremely unlikely that a true optimum or self-optimizing system can be constructed in the minimax sense, i. e., one which

will be optimum with respect to the most perverse future that the system will ever have to face. What must be done is much the same as was done in the random search scheme; that is, to devise a procedure that will work well in some cases and to apply the scheme to real problems hoping for the best. The great advantage of the random methods of search was their flexibility in application, varying from the crudest method guaranteed to work given time, to the most sophisticated based on a great many assumptions about the function. The Dynamic Extremal Control will be a method for realizing an adaptive system with similar flexibility using the search theory outlined in the previous chapter.

3/2/1 Adaptation versus Design

One way to think of adaptation is as a progressive self-design. An adaptive system can be one with variable parameters which are adjusted so as to improve the performance in response to the actual environment faced. A typical example of an adaptive system is a standard controller and the technicians who install it. The standard controller cannot be designed completely, since the parameters would not be appropriate for every process to be controlled. When it is put into service therefore it must be tested and adjusted. This is obviously an efficient procedure if (1) the standard controller is capable of good performance after adjustment, and (2) the general character of the controlled process does not change too much (at least between readjustments of the controller).

The adjustment process is the distinguishing feature of this particular type of adaptive system and a random scan or other computation technique might be used for the adjustment. This is a complicated form of feedback involving necessarily predictive aspects. In the type mentioned above it takes the form shown in Figure 3-1.

In mechanizing an adaptive system of this kind, there are two extreme situations which come to mind.

(1) The first extreme case involves the control of systems that are very difficult to model for one reason or another. An example is the supersonic aircraft which must have controlled maneuverability under conditions of speed, air density, etc., which may vary widely. The variations in the environment cause corresponding variations in the response of the craft to manipulations of the control surfaces. A fixed control system designed for such an aircraft must be fairly sophisticated to take account of all the significant environmental variables. A great amount of experiment (flight testing) must be done to determine the effect changes in the environment have on the response of the plane.

An adaptive system does not face nearly so formidable a task, since it needs only to measure the response of the system and adapt the controller parameters to it. The controller is thus automatically redesigned for successive responses, and the environmental reasons for these responses are quite irrelevant. There exists the fascinating possibility that the adaptive system will still function under conditions not conceived in design, e. g.,

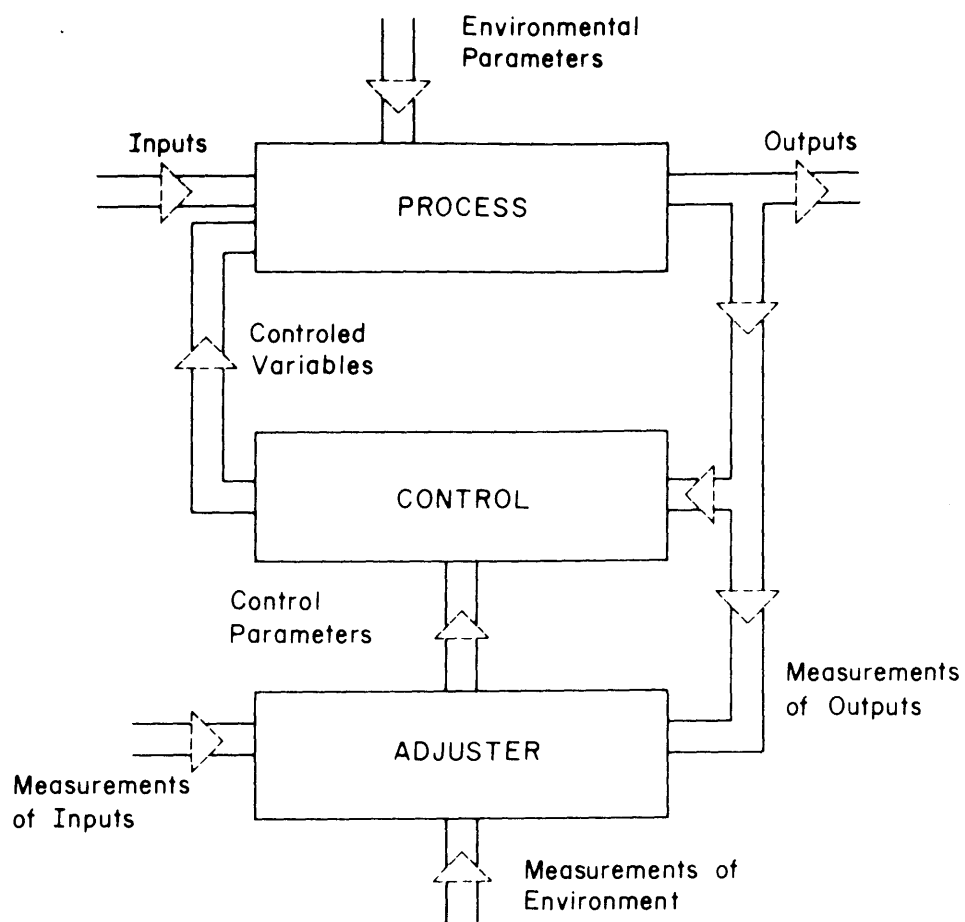


Figure 3-1

One Type of Adaptive Control

an adaptive autopilot might be able to maintain control even if some of the control surfaces of the airplane had been destroyed.

In the situations where the model of the system is not readily available, the straight-forward way to adjust the behavior is to run experiments on the real system at various parameter settings until a good set is found. This is after all just the procedure followed by organisms in nature. It must be evident, however, that this procedure has severe limitations. If there are any number of parameters to be adjusted, it must take a long time to search the parameter space, during which time the system itself may be changing significantly. Also many experiments may well be quite disastrous.

One way out of the difficulties associated with direct experimentation on the system is to somehow make a computer model of the system and then experiment on the model. Extensive experimentation could be done in adjusting a model controller to be appropriate for the model system. The need for high speed in these computations is evident.

(2) Another extreme would be a system which was fairly easy to model but too complex to analyze in detail at the design stage. A complex of inter-connected simple parts fulfills this condition.

Adaptation can be very efficient under these conditions when compared to a design procedure. For example, if one had a steady flow system with n inputs, m controlled variables, and some criterion function to be maximized, the design of an optimum controller could consist of a tabulation of the m optimum settings of the controlled variables as a function

of the n inputs. Now finding and storing m n -dimensional functions can be a formidable task, even for rather small m, n .

On the other hand, an adaptive way to accomplish the same result is to build a computer model of the system, feed in the actual inputs and maximize the criterion by computer experiment among the controllable variables. This involves one m -dimensional maximization at a time. Obviously, the first procedure can result in nearly the optimum behavior, since only the access time of the storage is involved before the optimum values are known. The adaptive scheme relies on fast computation for its success. The adaptive scheme does have the advantage of flexibility, since changes in the system need just be incorporated in the model.

In going from this static problem to the simplest dynamic one, i. e., when there is a step change in the inputs which can be expected to persist for a long time, the question no longer involves just finding the optimum steady state but also the method of achieving this state*. The memorization during design of the appropriate time functions for each controlled variable for every input and initial dynamic state is evidently not practical for even very simple cases. Here an adaptive controller is indicated (or for economic reasons, a fixed controller).

* Under steady conditions the optimum behavior may not even be steady but may well be periodic.

3/2/2 Examples of Adaptive Systems

Biological Systems

In any study of adaptation the origin of the term in the biological sciences must surely be studied. Since the time of Darwin at least the concept of adaptability has been central to the theoretical notions of biology. While it has been evident for some time that man could construct machines that have many characteristics in common with living beings, the search for the essential differences between living and non-living systems has continued. Animals are complex to a degree not yet found in a machine, but more importantly, the adaptive, purposive character of living systems has seemed to set them off from nonliving devices.

Although it is hard now days to find any excitement in the old mechanist-vitalist argument (see Wiener¹, p. 56), it is perhaps worth noting that mechanical adaptive systems can clearly be constructed which have behavior patterns very similar to those of animals. It is becoming increasingly evident that causally determined devices can achieve an over-all teleological behavior. Despite the immense complexity of living creatures, some of the mystery associated with life has certainly been dispelled.

One ought to note that adaptation or adaptability are terms that play a role in theoretical biology analogous to the term stability in dynamics; in each case what seems like an intuitive and evident definition

in general can be made precise (mathematically) in a surprising number of subtly differing ways. Perhaps then we will grow to distinguish between these shades of meaning by referring, for example, to "adaptability in the sense of Darwin" as we now refer to "stability in the sense of Liapunov".

As recently as 1950 G. Sommerhoff tried to use adaptability as the touchstone of life and to render the term precise, (Sommerhoff). His purpose was to show how a causally determined system could exhibit the teleological behavior typical of living entities. In words, his definition of an organism adapted to an environment with respect to some goal ran something like this: if the environment plus the organism were state determined, then an adapted organism would produce a response which would insure the attainment of the goal at some future time for a whole ensemble of possible initial states. The main point can be seen even in this simplified statement, i. e., that the adapted organism would produce appropriate responses for a range of possible environmental conditions. He imagined virtual changes in the initial conditions and corresponding virtual changes in the response. These changes would not necessarily ever occur.

Thus a bird adapted in this sense to pecking at seeds can successfully peck at a seed placed at any point within some region. Also the homeostatic readjustment of the carbon dioxide level in the bloodstream can occur for many different amounts of excess carbon dioxide.

With this definition of a system adapted to its environment, there are a great many adapted systems. A damped pendulum, for example, will attain the goal of no motion for any initial condition. A thermostatically controlled heater can return the temperature of a room to a set value for a range of outside (constant) environments. And of course the systems studied in classical mechanics have the goal of minimizing the time integral of the difference between kinetic and potential energy.*

It is a curiosity of the Sommerhoff work that it is merely an analysis of the adapted state with no discussion about how the organism could have attained an adapted state or in what manner the organism could possibly generate the appropriate responses. A modern answer to the latter query would certainly be that the general idea of feedback seems to be used to generate the appropriate system responses. Feedback control systems certainly are designed to promote just the sort of adapted behavior discerned by Sommerhoff.

It is interesting that Sommerhoff based his attempt to distinguish between living and nonliving systems, which can both be called adapted, on "epistemically" independent or dependent variables. To illustrate rather than define his distinction, he imagines that in nonliving adapted systems

* It is quite evident from the two possible formulations of classical mechanics, vectorial (causal) and variational, that a causal system may well also be said to behave teleologically in some sense.

the forces which assure the accomplishment of a goal (e. g. , the restoring force on a pendulum) are rigidly related to the state of the system, while in a living organism they are not (at least in principle). But if an organism really does employ a rigid feedback control, then the distinction disappears.

Thus it would seem in modern terms an adapted system is one which has an effective control system.*

This definition of an adapted state is really teleological, since one can hardly imagine determining whether a device has or seems to have an effective control system without having in mind some goal which the control system seems designed to attain. One such goal is the achievement of a steady state. A homeostatic mechanism has this as a goal and it is adapted if it is stable. Thus the systems discussed by Ashby (ultra-stable and multistable systems) are special cases of adapted and self-adaptive systems (Ashby).

Whether or not Sommerhoff's definition of an adapted system is appropriate, we will be concerned here more with adaptive systems and the process of adaptation. Certainly almost all biological entities are adaptable to a degree. Obviously, human beings and all organisms start out in life with a very small number of programmed responses to the world (instincts and reflex actions), but in time the more complex learn to make

* The control system need not even be a feedback one -- it could just be programmed from past experience (instinct).

quite complicated responses to the environment in order to attain their goals. In human beings even the goals are largely learned, not inherited. To the extent that animals possess this sort of facility for adaptation, they are commonly ranked as higher animals. In the cases of such higher animals, it is unthinkable that information necessary for performing all the complex operations which might be required during a lifetime should have been stored in their brains at birth. It seems on the contrary not so difficult to imagine the animal capable of adapting his behavior to whatever environment in which he may find himself.

On the other hand, lower animals do have their responses almost entirely programmed at birth. In many situations the adaptability of these animals can be seen on the species rather than on the individual level. As an example, the killing of any individual fly with DDT is found to be no great feat, but the extermination of all flies is a formidable task. Not only are there great numbers of flies throughout the world, a small number of which could repopulate the globe very quickly, but it is found that quite quickly a DDT resistant strain can be developed by the species. In this case it is profitable to consider the entire species of flies as a vast adaptive system.

In a similar way the exponential increase in knowledge about the universe which is often attributed to man is certainly not due to a change in the characteristics of individual men (at least since the dawn of civilization) so much as it is due to the learning and adapting of mankind or society.

In a sense, society in general is an adaptive mechanism which through storage and communication of information between its subsystems is able to achieve much more than the subsystems could achieve in isolation.

Man-Made Adaptive Systems

In recent years the literature in the field of control has been full of reports on adaptive control systems. It would certainly be a monumental task to give a complete bibliography on this field, so we have simply selected a number of articles of interest to mention in the present bibliography (Aseltine, Draper, Gregory, Merriam, **ФЕЛДБАУМ**).

Some general statements can be made about all man-made adaptive systems. In the first place, it must be kept in mind that while the study of biological systems may be very instructive, man cannot expect to merely copy these systems in any detail. This is because the biological system is apparently made up of very large numbers of rather simple parts, while a man-made system must be composed of much fewer, but possibly more complex parts. It is often desirable to model the over-all behavior of biological adaptive systems, but this must be accomplished by means quite different from those used by living creatures.

In the biological world it is hard to believe that any but the very highest form of life is capable of performing imaginary experiments to attempt to predict the results of any actions. This question is tied up with the question of self-consciousness, and it is commonly assumed that only

man among the animals has a mental image or model of himself. Without the possibility of mental experiments an organism is forced simply to try a course of action which has worked well in similar situations in the past, and to wait for the results. If the results are not satisfactory there may be a chance for another try, and usually that try will be a random variant of the first. Even a human being if suddenly faced with an inimical environment may go into a state of panic which is characterized by very elemental responses, some of which could be shown by rational means to lead only to disaster.

There is a similar general distinction between two classes of man-made adaptive systems. In the first class, there is very little reliance upon a model of the system, and the adaptation is carried out on the physical system itself. In the other class, there is reliance upon a model of the system, and under changed conditions a prediction can be made about the effects of changes in the controllable quantities. Each class has some advantages and some dangers, and we shall examine prototypes of both classes to illustrate them.

Perhaps the most obvious example of the first class is the optimizing control of (Draper and Li). This type of control, often called extremal control, attempts by adjustment of a physical parameter to optimize an essentially static function. The example quoted in the original paper was the problem of optimizing the over-all efficiency of an internal combustion engine by the manipulation of a fuel/air mixture control.

The great advantage of such a system is that no matter what has caused a change in behavior of the system, the optimizer will readjust its parameter to at least a relative optimum as long as the change persists for a long enough time. In a sense then such a system relieves the designer of the problem of considering all possible operating conditions.

There are, however, some rather severe restrictions to this sort of an adaptive system. Only a very few parameters can be adjusted (often only one) because the adjustments must be done slowly enough so that the physical system can react and be measured. If the system cannot find the optimum adjustment quite as quickly, then the adaptive searching of the system may not be doing much good. It is very likely that such a system cannot afford to spend time searching for the absolute optimum but must always settle for a relative optimum. Even then the system must always search nonoptimum points neighboring the relative optimum to make sure that the location of the relative optimum has not changed. The search is thus paid for directly out of performance, and hence such a system may or may not be profitable compared to a well designed fixed system.

Another example of such a system is the adaptive autopilot (Gregory), mentioned previously. In this case the parameters of an autopilot are adjusted so that the plane-autopilot system responds to a test disturbance in the same way that an ideal model would. Incidentally, the model

used as a reference often is adjusted so that if the plane-autopilot system behaved just as the model, the behavior would be called satisfactory or reasonably good but not necessarily optimum in any absolute sense. This is one way out of the philosophical difficulties in trying to design a truly optimum system. The disadvantages of this type of adaptive system appear in a critical form in the autopilot example. For one thing, the measurement of the effectiveness of a parameter setting must wait for the completion of a dynamic response. (In the previous example, the response was essentially instantaneous). Thus in this case the searching cannot be done by a continuous scanning but rather must be accomplished by discrete steps with a definite time lag between them. This means that the rate of adaptation and the size of the search space are rigidly tied to the dynamic response time of the system.

A danger is that inadvertantly an unstable adjustment might be attempted under an unusual condition. Since an unstable trial might well prove disastrous, the possibility of such a trial must be guarded against beforehand. This could lead to the use of only conservative searches which could not approach the absolute optimum under some conditions. Thus it can be seen that this form of adaptive system is generally limited to slow searching of small spaces for relative optima with an a priori attempt to restrict the search so that in no conceivable situation will the search result in a disastrous trial.

A way to circumvent some of the limitations of the first class of adaptive systems is to allow the system to have a self-consciousness, as it were, by the incorporation of a model of the system. This model of the system need not actually be present physically, but could have been present in the design stage. For example, if the optimum fuel/air ratio for an engine was found to depend mainly on air density, a chart could be made with which an operator could adjust the fuel/air ratio after reading an air density meter. In this case the system can adapt quickly to an absolute optimum without a search on the actual system, as long as the results based on the previously tested model remain valid for the actual system.

In order to avoid storing a large amount of data, it is possible to build a model of the system into which measured environmental conditions can be fed. The adaptive mechanism can then experiment on the model to find the optimum setting, with no fear of disastrous trials and probably at a very much faster time rate than could be accomplished on the physical system. In the airplane example if a number of environmental quantities could be measured and fed into an electronic model of the plane, the adaptive mechanism could then rapidly search the parameter space of the model autopilot until an optimum setting was found and then readjust the physical autopilot to the same setting. All of this could be done without disturbing the airplane itself.

It is clear that all model systems as described above are fallible in that unexpected changes in the system can occur which may not be taken into account in the model. This could be true for any environmental change which is not monitored or any change in the system itself which was not foreseen (for instance, deterioration with age). Thus the type of model system discussed above corrects the faults of the first class of adaptive systems but may lose the one great advantage of such systems.

As human beings we are often faced with the same sort of question. We may decide to proceed as our preconceptions about the world indicate, or we can try several alternatives somewhat blindly to see which one actually turns out best. There is a third alternative and that is to refine our notions about the part of the world in which we are interested by experiment, and then make our decisions based on imagined or modeled experiments on the refined model.

In adaptive system language this process is called model updating or process identification (Haddad), and it seems to represent the most advanced conceptions in the field of adaptive systems when it is used with a self-optimizing scheme (Kalman). Although a model system of adaptive control with a flexible model repeatedly refitted to the process could probably be shown to be the most powerful method possible, one must keep in mind the great complexity of the system that is being discussed. When the judgments about computing systems are mainly

economic, many questions of theoretical effectiveness may become purely academic. On the other hand, if a near approach to optimal behavior involves a large payoff or if an adaptive system can perform a task which needs to be done and cannot be done by other means, then these considerations may have been fruitful.

In the next section and later in the last chapter, ways are suggested to use the search theory of Chapter 2 to construct an adaptive system. The method is of great generality and need not be particularly expensive in many cases.

3/2/3 Dynamic Extremal Control

In this section a term is coined and an adaptive control scheme is proposed. In the adaptive control field, which is already crowded with proposals, we feel that this proposal may be useful for several reasons. In the first place the method addresses itself to the most basic problems facing an adaptive system without losing generality by being overly particular. Secondly, the method explicitly considers the adaptive control problem as a dynamic optimization search problem, and thus the methods of Chapter 2 are applicable. Thirdly, such a method could be realized effectively in a number of ways, one of the most attractive of which would be with a high-speed analog computer. The use of an analog computer for the construction of adaptive systems will be discussed in Chapter 4.

The term Dynamic Extremal Control is meant to imply that a search for an extreme is carried on by the controller, much in the manner that

the Extremal Control of Draper and Li searches for an extreme. However, by the use of a high-speed model, we propose to conduct a search of a true dynamic function; that is, a function which varies in time and may involve the solution of dynamic equations in order to determine the value of the function when the search parameters have been set.

There are two basic problems which must be attacked in the proposed scheme. The first involves keeping a model of the system up to date. This must be done, since one of the attractive features of an adaptive control is a readjustment of the control to suit a changed system. The other phase of the problem is to find the optimum pattern of control given a good high-speed model on which to experiment.

It seems clear that it is foolish to attempt to design a universal model or a universal controller that can simply be specialized to suit any conceivable system. Occasionally, however, one comes across an article which effectively attempts this, and in these cases it is hard to see how such systems can help being extremely sensitive to noise. After all, if the model or controller attempts to work too closely with noisy data, it may be making crucial decisions on insignificant features of the data. In the proposed scheme all the engineering experience with the system that may be available can be incorporated. For example, it can practically be guaranteed that the performance of a given controlled system can be at least somewhat improved by a minor variation of the parameter of a fixed controller which is known to be fairly effective. More

radical variations from usual practice can be tried and incorporated if such variations yield further improvement in performance.

In the making of a good model of a complex physical process, it is clear that science has not yet supplanted art. While almost anyone with a proper technical background can invent a very complex model for a system, only one with skill and experience is likely to be able to sort out the most important quantities in order to form a simple but adequate model. It is true that the behavior of many very complex systems can be modeled with very simple models if the parameters of such models are suitably adjusted. The range of possibilities is large, from the large system whose components are numerous but well understood, to the cases in which the structure of the relations is unknown. Indeed, there is even progress recently in the modeling of industrial dynamics. In almost every case one finds that as the systems are better understood, there can be more approximations made which do not deteriorate the prediction capabilities of the model but do make it more manageable. In particular, one is frequently not interested in many characteristics of the system so that any model that will generate the quantities of interest with reasonable accuracy is valuable whether or not other details of the behavior are modeled well or not.

We propose to construct a high-speed model along the following lines. The model will be based on experience with the physical systems and will incorporate a set of variable parameters of a general sort.

With a proper resetting of these parameters, the model should simulate the system behavior under various environmental conditions, under the influence of inputs of varying characteristics, and under as many disrupting influences on the system as possible.

Two aspects of this model approach should be emphasized. First, it should be possible to match the model to the system even when the system has changed for an unpredicted reason. In this regard the model cannot be expected to match all aberrant behavior of the physical system, but perhaps the match will still be good enough to be useful in many cases.

Secondly, one must be imaginative in the use of parameters in modeling. For example, the amount of noise injection is clearly a parameter which might be important. Also the use of discrete parameters to indicate the presence or absence of subsystems is legitimate. Simple analog techniques for this purpose will be shown.) In fact, a parameter which takes on one of a series of discrete values can determine which of a set of completely different alternate models is to be used.

In attacking the problem of which model or which parameter setting is appropriate at any instant of time, there are two possibilities; either a change in some quantity which enters the model directly can be measured or changes can be noted indirectly by experimentally matching the model to the system. Certainly there will be mixed cases in which some effects can be incorporated directly and others cannot. In any case it will certainly be wise to check the model against the actual physical system it

is supposed to represent fairly often. Although there have been any number of schemes proposed for model matching which could be used in appropriate cases, the most direct one possible is proposed here -- a simple search in the parameter space of the model for that set of parameters which best models the recorded recent past history of the physical system. Obviously, the length of past history to be used in comparing the model and the system is a compromise between the goal of detecting changes in the system quickly and comparing the system and its models over a significant length of time. There is a definite limit to the speed of adaptation, for even with infinite searching speed no change in the system characteristics which is not directly measurable as a changed parameter can be recognized until the change shows up in the behavior of the system. To make a valid estimate of the change in behavior of a system generally requires a finite length operating record, and this is the history to which the model is adapted.

Another situation which may occur is that the system characteristics may change during a time when there is not enough variation in input and output to determine the model. For example, if the system changes during a time in which both the input and the output are nearly steady, then any stable model will seem to be as good as any other under any criterion. In this case there is nothing to do but disturb the system so as to record some significant input-output information from which the best model may be selected.

One might object to this disturbance of the system on the grounds that if disturbances are to be tolerated, one might as well perform adaptation by direct experimentation on the system. In some cases this could be a valid objection, but in general a great many models can be tested using single recorded disturbance, since it is postulated that the model is fast in comparison with the system. Thus shortly after the disturbance, a valid high-speed model should be available on which to experiment. The proposed scheme then can do rather extensive searching on the model to determine, for example, the optimum setting for a controller. If such an optimizing search were carried on directly with the real system, it would have to proceed at the time scale of the system and probably would require a disturbance of the physical system for every trial point. What our proposed system does is to substitute two high-speed searches; one for model matching, and one for optimization for one low-speed optimization search.

Given a valid, high-speed model of the system, the question of the optimization search is still not trivial. Although the capability exists of trying many alternatives, i. e., searching a large space, the philosophical difficulties associated with the definition of the optimum are still valid, for the statements about the optimum behavior are statements about the future, and the model is of definite but limited value in the prediction of the future.

One possibility is to proceed as in the autopilot example to specify that the over-all performance of the process and controller must be

similar to the performance of a reference model and ask for those settings of the model controller which best provide such a behavior. In this case the advantage to be gained from the Dynamic Extremal Control is that the search can be done at high speed and by putting test inputs into the models rather than into the system. After the best control parameters have been found, the physical controller can be readjusted and the process repeated. With the use of a model the scheme can search for absolute optima, not merely relative ones, and can if need be search regions interior to an unstable region.

Another straight forward approach would be to ask for the optimum strategy for the controller for the recent past and use this for the near future. The optimum relative to the past can be well defined, and its use in the future is useful if there is a rough continuity to the time variable aspects of the problem. Obviously in implementing this idea, one must devise a system which will be sensitive to sufficiently slow and persistent changes and not to high frequency changes which must be treated as noise and must not unduly influence the adaptive scheme.

The question of what a good controller should have done in the past is well defined but difficult of solution. If one simply asks for the time history of all the controlled variables which would have resulted in the best performance, one may not discover a useful rule for the future. In fact, the problem of finding such an optimum time history is really a variational problem equivalent to a search in an infinite dimensional function

space. Of course, if only finite accuracy requirements are considered and perhaps an upper limit is placed on the frequency content of the signals, the problem may not be quite so bad, but it would seem not the best way to attack the problem. Rather what is usually of interest is not the actual time history of the controlled variables but rather the rule which generated them. One way to specify the rules is to specify a realizeable controller circuit which will generate the time functions.

There are many processes which could be controlled very well by a fairly standard type of control if it were continually adjusted. In these cases it is possible by experience to reduce the problem to a readjustment of a manageable number of parameters which could be done rapidly on a high-speed model. Again the parameters could be very general in nature, even dictating the use of entirely different control schemes at times. The point is that it is not necessary to search among all possible means of control of a process if there is engineering experience with the system.

An intriguing possibility is that an adaptive system, which adapts to the recent past history, may be adapting to stationary aspects of inputs, environmental conditions, or the systems characteristics which need never be recognized by the system designer. In many proposed adaptive systems the designer picks some quantities which can be measured, and the system is constructed to be adaptive with respect to changes in these quantities. In Dynamic Extremal Control if adaptation can be

accomplished using recorded history, in effect the system itself chooses those aspects of the history which are important. In this sense the schemes we propose are adaptive in a much larger sense than many adaptive controllers which have been proposed previously.

There are occasions when time functions might be considered explicitly rather than as they might be represented by the parameters of a device which produces them. An example might be the case in which the system is required to execute a given transition from one state to another in an optimum way. Specific examples include the start up and shut down of a fairly complex system. In these cases the appropriate time history of the set points of the controllers can be studied. If the set points of j controllers are to be given at each of k intervals of time for the future, a jk -dimensional search problem is involved. Dynamic Programming can sometimes be used in such a situation if the appropriate sort of computer storage is available. Usually, since a number of one-dimensional computed functions must be stored, a digital computer which is fast and has quite a bit of storage must be used. See (Merriam). Without Dynamic Programming the jk dimensional search is rather difficult to achieve in a short time, however, some of this sort of work can be done in the design phase so that not all possibilities need be tested. For example, the device might test a series of stored programs for changing the set points to accomplish a given transient and use the model merely to pick the best one and to warn of any danger in attempting the transient based on the stored programs.

This brings us to a final unique use of a high-speed model adaptive system. The model can be used to predict the future behavior of the system under any particular control scheme. This might be very important if the process had a possibility of drastic misbehavior. A complex atomic power generation plant might, for example, get into an unexpected mode of operation or state, perhaps due to a malfunction of some part, which would lead to a disaster predictable by the model. If such a dangerous tendency were predicted, there might well be time to shut down the unit before it was too late. With this predictive feature this type of adaptive control can avoid at least some of the noted long-run difficulties which arise when only short run optimization is used.

CHAPTER 4

THE ANALOG COMPUTER IN OPTIMIZATION STUDIES

This chapter demonstrates that although the analog computer has been used for many years in optimization studies, it has not reached its potential usefulness. Mainly this is because the analog has been used in direct communication with a human being in contradistinction to the digital machine which very early in its development began to communicate only digested results to human beings. Thus the analog computer while having potentialities for high-speed operation far in excess of the digital machine's reasonable potential on certain problems, has never been developed in this direction. We shall show how the methods of Chapter 2 are singularly suited to analog computer applications and thus how the analog computer, properly developed, can be usefully incorporated in the systems of Chapter 3.

4/1 SUITABLE PROBLEMS

It is probably hardly necessary to discuss the fact that some problems are much more suited to electronic analog computation than others. Leaving aside questions of speed, accuracy, and cost, which shall be dealt with separately, the main question is whether the problem can be formulated in one continuous independent variable. The most natural problems are initial value problems involving ordinary differential equations with time as the only independent variable. Difference equations, partial differential equations, or algebraic problems can be solved on an analog machine in principle, but the procedures tend to be awkward and

inaccurate.

At any rate, an analog computer is eminently suited to the solution of dynamic equations involved in control problems, and it is these that we shall primarily address ourselves.

4/1/1 Accuracy

The maximum accuracy of an analog machine is fixed by the design of its components and is not expandable as in a digital machine. This fact has implications which are sometimes mistakenly given as independent limitations of the analog machine. For example, the necessity of good scaling for all the variables is an outcome of the attempt to attain the maximum accuracy for the entire system. The same problem would occur in a digital machine if an attempt was made to maintain the greatest possible accuracy in the final answer of an involved calculation given the accuracy of the individual calculations. Also the upper and lower bounds on the voltages representing the variables do not in themselves represent a limitation of dynamic range. After all, these bounds are finite numbers just as 10^{32} is. However, these bounds together with the limited accuracy of the machine combine to limit the range by fixing the ratio of the smallest discernable number to the largest possible number.

Thus there are distinct limits to the problems which can be handled by a given analog machine. However, one should not be too facile in making decisions about the suitability or unsuitability of analog computation. The mere statement that a parameter must vary through a range of

100,000 to 1 does not rule out the possibility that the effect of the parameter saturates so that virtually no change is observed at ratios of greater than 50 to 1. On the other hand, very small ranges of variation of some quantities may affect the solution so drastically in certain ill-conditioned cases that high accuracy is completely necessary.

It is also not clear what the absolute limits of accuracy of analog computer components are. Of course economic considerations combine with the state of the art at any time to give a practical limit to the accuracy, but it would be presumptuous to give a firm limitation to the accuracy which could be obtained if a sustained effort were made in the direction of accuracy improvement.

The accuracy problem is further complicated in the sort of problem that has been concerning us by a trading between speed and accuracy. Even in a design situation a fast search of a function of a large number of variables may reveal that the function is so flat that an accurate search would not be worthwhile. In the time variable search problem there is, as we have pointed out, a direct correlation between search speed and the maximum possible accuracy. In time-variable cases it is possible that an analog scheme which could solve simultaneous time differential equations much more rapidly than a digital machine could prove to have the greater accuracy.



Room 14-0551
77 Massachusetts Avenue
Cambridge, MA 02139
Ph: 617.253.2800
Email: docs@mit.edu
<http://libraries.mit.edu/docs>

DISCLAIMER

MISSING PAGE(S)

122

4/1/2 Speed

It has been known from the beginning of analog computation that solutions to complex problems could be attained very rapidly on analog computers even though the component parts of such a machine might not compute very rapidly. The reason of course is that the computations go on simultaneously or in parallel for all parts of the machine rather than in serial order as is done in digital schemes. Furthermore, the analog computer usually works with continuous functions of time, and need not use a great number of sample points to represent such functions. The price that is paid for such fast solution of complex problems is that more complex problems require more complex computers. On the other hand, a serial operation machine can, within limitations of storage, handle a problem of arbitrary complexity, but the time for solution will increase for more complex problems.

Obviously, there is no reason in principle why analog machines should be parallel operating and digital machines serial operating, but there is the very important matter of cost. Analog computers are not particularly expensive devices, and adding to them to increase their problem-solving capabilities does not represent a major cost. On the other hand, high-speed digital machines are expensive, and modifying them to do more than one operation at a time would represent a great increase in cost.

Interesting comparisons have been made between analog and digital computers for simulation of the dynamics of airplanes in training devices. These devices have been built with analog computations using very slow components such as servo-multipliers. It is clearly almost possible for the newest digital computers to be fast enough for this application and yet compete economically with the existing analog devices. But this is a comparison between a highly developed digital computer and a very slow analog computer.

Almost ten years ago D. M. MacKay, and certainly others, had already built analog computers which were much faster than today's commercial units (MacKay). There are several reasons why there has not been a continuing emphasis on speed in analog computation. In the first place, many analog computer installations are used for real time simulation. In these cases the speed of the analog computer is rarely taxed, although as the aircraft simulation study has pointed out, the speed capabilities of a serial operation machine might be severely taxed.

In the second place, there has always been a desire for direct human interaction with analog computers. Most optimization problems are carried out by a human being who adjusts the parameters of a problem which the computer solves repetitively. In this case if the solutions are repeated at much over 10 c.p.s., the effect of a parameter change appears instantaneous to the human. If an adjustment has not taken place for $1/10$ of a second or more, the machine is simply repeating calculations.

McKay found a use for higher repetitive speed in a curious way. He arranged to have the machine vary one or more parameters through a finite set of possibilities, compute the corresponding solutions, and then repeat. The speed of the machine was so high that the entire sequence could be repeated fast enough for human persistence of vision to record a steady pattern of solutions on an oscilloscope face. Again this is a peculiar emphasis on the direct observation of a human when a high persistence oscilloscope or camera would have allowed very much slower speeds to suffice.

It is our proposal to emphasize and make use of the dramatic high speed potentialities of the analog computer in the case of search problems in which speed may be completely essential. To do this the luxury of feeding the human all sorts of redundant and uninteresting information must be given up. In an optimization problem, for example, the machine must decide when one solution is better than another. It is still possible to feed useful information to a human, but it must be digested. An interesting example that we have done on standard equipment at M.I.T. is to perform an optimization problem by random means and to display every solution as it is run and a repeating display of the most nearly optimum solution that has been discovered up to any given time. Even at 10 c.p.s. the trials go on at bewildering speed, but one can follow the history of the best-so-far solution as it rapidly improves at the beginning and then changes more and more slowly until the solution is optimum within the accuracy of the computation.

This sort of computation does not repeat calculations until the machine has run out of accuracy, i. e. , until the search space has been covered so densely with trial points that neighboring points are indistinguishable.

Using this sort of approach problems can be solved which require great speed, and the full speed potential of the analog computer can be utilized. Even standard computers can, with minor modifications such as changing from mechanical relays to electronic ones, easily run up to 100 c.p.s. , and 1,000 c.p.s. is certainly achievable. It is hard to say just how high the solution speed could go if an effort were made to design fast components. It is not inconceivable that these entire solutions with varied parameters could be run off at a megacycle rate. Previous chapters have demonstrated that some problems require very high speed, and we shall show in the remainder of this chapter how simple methods for utilizing such speed can be. If there are problems which badly need to be done in this field, the analog computer is worth a great amount of study, since the speed advantage over digital machines can be very great.

4/1/3 Cost

The question of cost of an analog computer for the automatic solution of optimization problems is far too complex and speculative to permit a definitive discussion here. There are a few general considerations to keep in mind, however.

At one end of the spectrum of possibilities are those machines, yet to be developed, which are specifically designed for very high speed automatic operation. For some problems there would be no reasonable alternative methods of attack besides the high-speed analog computer. Thus the cost of such computers must be balanced against the cost of not solving the problems or by only partly solving the problems with whatever approximation methods may be available. Certainly many unsolved problems of this sort exist, but it is not clear what economic factors are involved. Such a machine might be used in the control of a complex process or as a general research and design tool for exploring the effects of many parameters which previously have had to be fixed, for example, by rules of thumb.

At the other end of the spectrum are the low-cost machines, assembled from existing components for the most part, which may yet do fairly sophisticated tasks. It is possible that an analog installation of this sort might be so inexpensive as to have no reasonable competitors. The cruder versions of random methods seem ideally suited to this sort of application.

In between these extremes complex cost questions arise. For example, digital computers, while almost always quite expensive, often can perform a great variety of useful tasks and thus pay for themselves in many ways. Most analog machines, while less expensive, are also more limited in the type of application for which they are suited. In the

middle ground in which great speed is not absolutely required and extremely low cost cannot be achieved, individual cases must simply be judged on their own merits.

4/2 LOGIC AND MEMORY

The additional equipment which must be added to a general-purpose analog computer so that it can be used for automatic optimization problems falls mainly in the area of logic and memory. The logic is used to control the flow of information within the machine, and the memory is used to record the inputs and the results of computations. Most analog machines have at least a primitive control structure for doing repetitive operations, and other logical and memory circuits can be fabricated from standard components. In this way, using almost no special equipment, experiments were run on the M.I.T. analog computer. However, special equipment can readily be designed which is particularly suited to the specialized jobs required for automatic optimization, and then the main part of the computer will be available for the problem in question.

It would be convenient to have a digital section of the machine with flexible logical units for establishing the timing of the basic computational cycle, performing logical operations during the cycle, and for some types of storage.

4/2/1 Advantages of Simple Random Methods

For at least two reasons the simpler versions of random search techniques seem appropriate for analog computers. The first is that only a limited accuracy can be expected from such a machine. Therefore, problems which require great refinement of an approximate solution will not be attempted. Rather the emphasis will be on the initial search, and only a few refinement stages can be worthwhile. It seems likely that there will never be a need for very highly deterministic refinement schemes, since the simple modifications of random searching such as narrow moving mean scan will be sufficient, and artfully used will be superior to the use of a highly deterministic scheme too early in the search. Indeed, overly deterministic methods must be avoided as the accuracy of the approximate solution reaches the noise level of the machine, or the scheme will not converge well.

The other reason that simple random methods are so attractive is that they can be arranged to put minimum demands on the logic and memory of the computers. Much of the logic of a problem is stored in the connection pattern of the elementary units of an analog computer, so it is desirable to get along with as little extra logical equipment as possible. The simple random methods require the absolute minimum in memory and logic. As an example, a wide scan is conveniently realized on a digital computer as a fixed grid scan, since this can be generated by counting procedures. To do this on an analog computer requires the same sort of logical

facilities as a digital machine. However, a random scan obtained by sampling noise sources requires practically no logic at all and has just the same effect. Similarly, a surprising amount of determinism can be introduced into a random scan with very little logical equipment.

In all of our techniques we have consciously attempted to reduce the memory requirements as much as possible by requiring that only the most powerful pieces of information be stored. Our emphasis on the storage of F^* and x^* is in accord with this goal, and our pattern recognition schemes again were conceived to require minimum memory facilities as well as a minimum of logical operations.

4/2/2 Techniques

Timing and Gating

A conspicuous feature of an analog computer which can perform automatic optimization is a provision for a more complex and flexible timing cycle and provisions for gating signals in response to simple logical operations.

The simplest timing cycle used for analog computer repetitive operations is set-run. In the set phase the initial conditions are set to prescribed values, and then in the run phase the system computes the solution to the equations involved. For our problems we must establish a set-run judge cycle so that after the solution has been computed there is time to make judgments about the solution. See Figure 4-1 for a typical cycle.

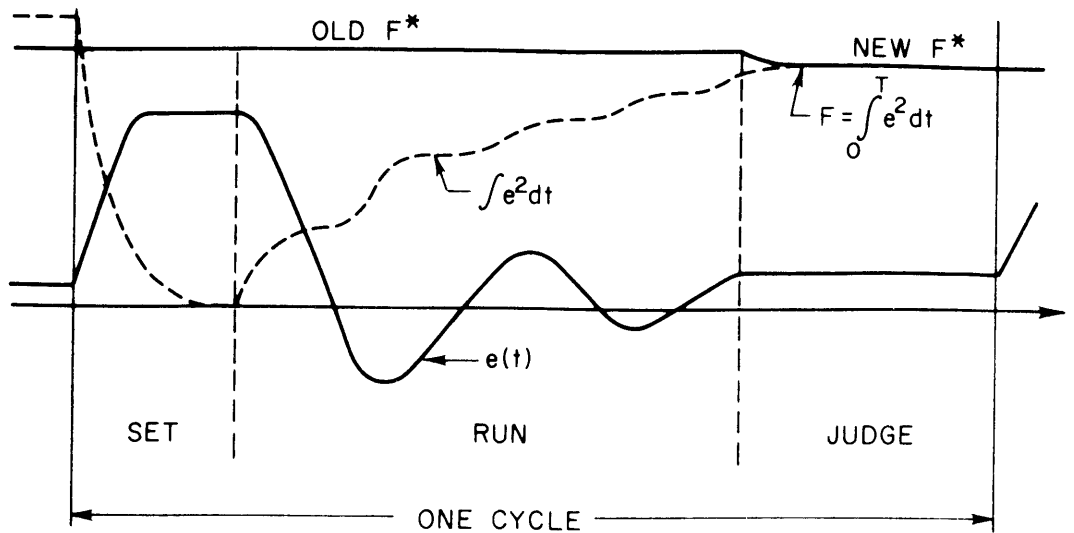


Figure 4-1

A Typical Analog Computer Cycle During a Static - Dynamic Search for a Minimum

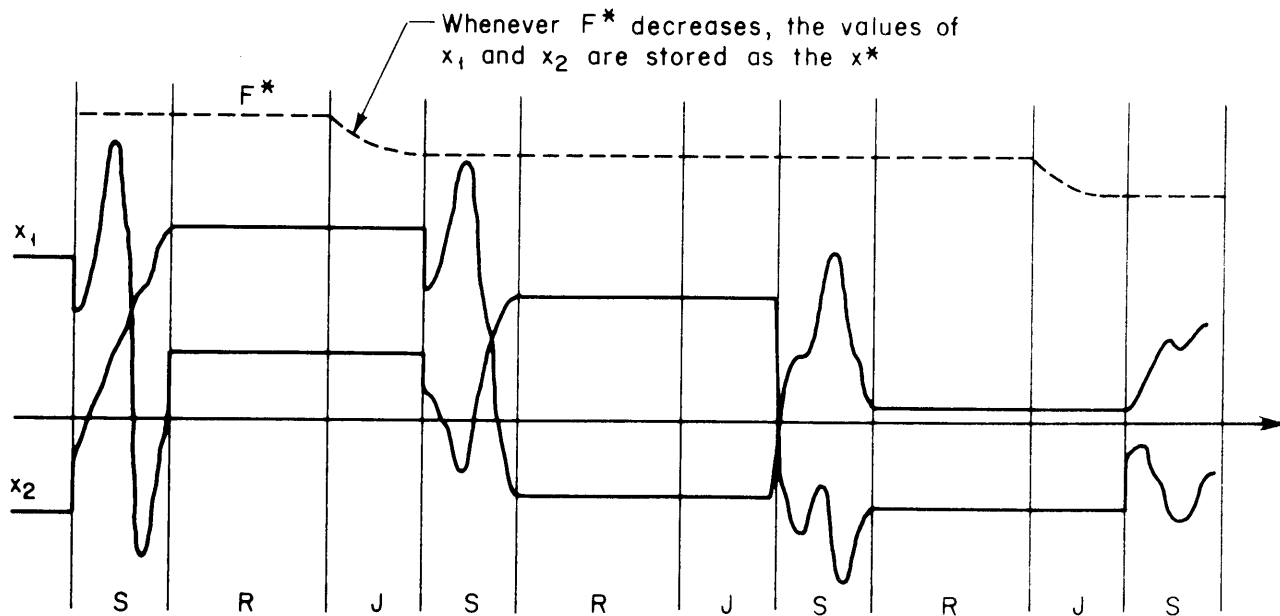


Figure 4-2

A Typical Record of a Minimization Problem

The parameters x_1 and x_2 are generated by sampling random waves at the end of the set phase.

The problem illustrated in Figure 4-1 is to find the set of parameters of a servo system such that the integral of the squared error of the system when subjected to a step is minimum. In "set" the initial conditions are established, and the trial parameters (not shown) are fixed. In "run" the solution is computed. In "judge" the function $F = \int_0^t e^2 dt$ has been evaluated and is compared with the lowest previous trial value, F^* . In the cycle shown $F < F^*$ so the present F replaces the old F^* , and the parameters which caused this lower value of F are stored in place of the previous values, x^* .

During the "judge" and "set" phases certain gating operations can go on. During "set" the parameters are reset, and the scan distribution can be changed. During "judge" the decision whether or not to store F as the new value of F^* is made, and corresponding judgments are made about x and x^* . See Figure 4-2. In the figure a wide range scan is accomplished by sampling and holding values of independent random waves. There are many ways to generate such waves, and in fact, desynchronized periodic waves of a shape corresponding to the distribution desired may be used. Even the timing of the sampling is not critical, although it is preferable to start holding the x 's somewhat before the "run" phase actually begins.

To generate a narrow moving mean search, one can simply sample a wave composed of the sum of x^* plus a narrow random wave,

$W_1 = x^* + w_1$. Thus whenever x^* changes, the center of the wave W_1 and

hence the scan changes. If it is desired to construct a scan with both wide range and narrow range aspects, this can be done by switching randomly between a wide wave, W_2 , and the narrow wave, W_1 , before sampling. If this is done for each variable, the result is as shown in Figure 4-3.

The comparison between F and F^* is critical for the success of the system. If this comparison is not done to an accuracy equal to or better than the accuracy of the machine, it becomes the limiting part of the computation. This comparison controls the gates which allow storage of F as the new F^* and x as the new x^* . Obviously, the "judge" phase must be long enough for the comparison and storage to occur.

Various other gating activities go on which are not linked to the computation cycle. For example, decisions to change the scan from a wide search to refinement technique may be automated using the perception techniques suggested in Chapter 2. In this case the mechanism for obtaining the trial values of x can be modified at any time in the cycle. Also the machine may be programmed to terminate the search on the basis of any useful criterion.

For a general-purpose machine a series of relay-like devices, comparators, and the like which can be interconnected to accomplish a variety of simple logical tasks are needed.

An example of the use of such gating and logical devices is the use of discrete valued parameters to study possible system interconnections.

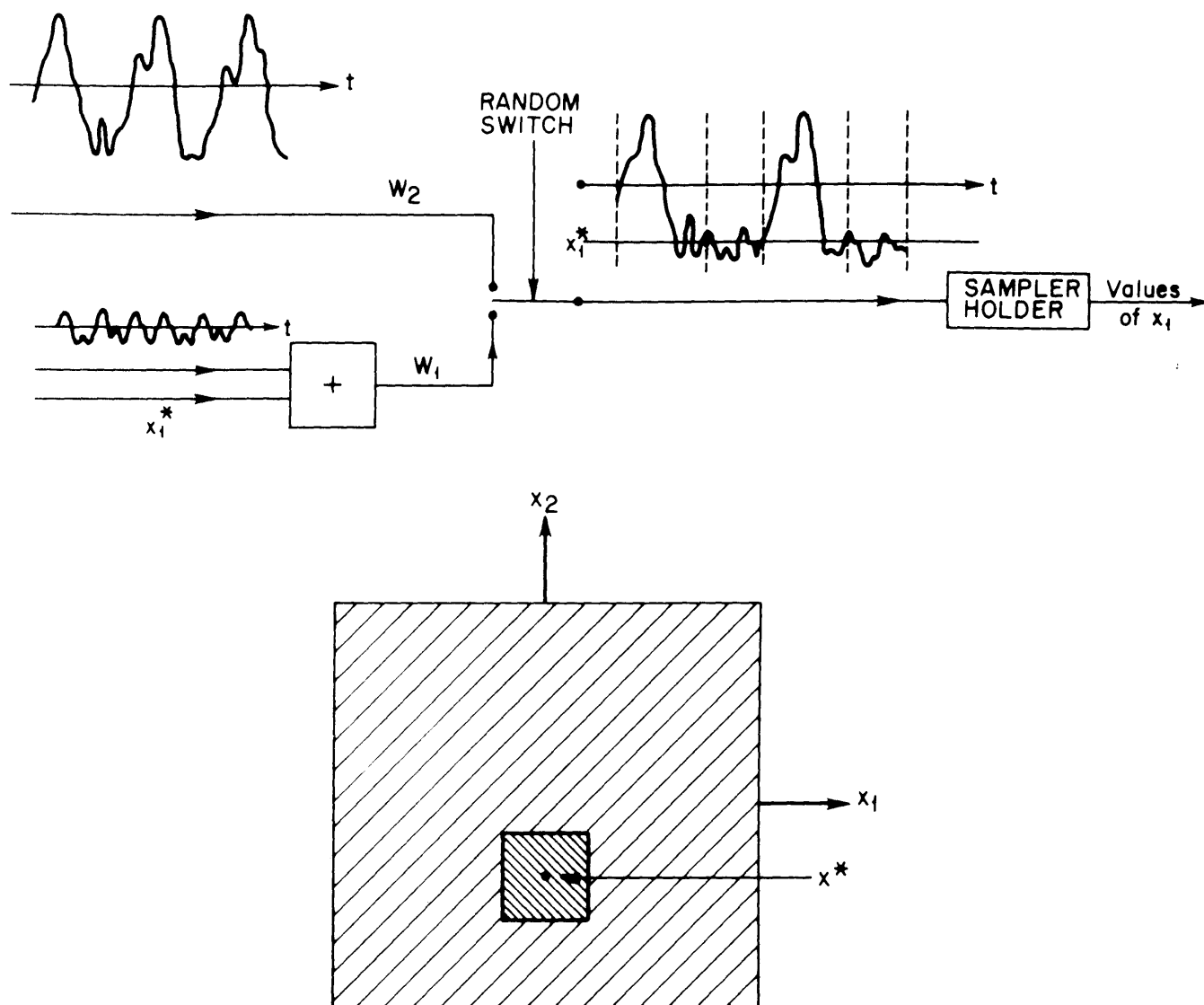



Figure 4-3

One Way to Generate a Scan Distribution with Both Wide Range and Narrow Range, Moving Mean Aspects.

Figure 4-4 shows how binary signals can be generated and used to interconnect various controllers to a system. Not shown are the parameters of each controller which could be scanned simultaneously.

Binary signals can be formed from the random (or semirandom) waves used for continuous parameters by running them into detectors. These detectors change state whenever the random wave crosses some reference value. See Figure 4-4(a). The level of the reference voltage will set the binary probability distribution. See Figure 4-4(b).

The sampled binary parameters are held constant during the run and judge phases and are used in a switching circuit to connect various subsystems to the main system.

As an example of such a switching circuit, Figure 4-5 is a diagram for the case of four controllers where the controllers all have the same probability of being connected into the system. The signals, a, b, c, are the binary parameters. The symbol  is used to suggest a mechanical double throw relay or the equivalent electronic switch. The fractions near the poles represent the fraction of the time spent at each pole or the binary distribution.

Thus without extra equipment the analog computer can be used to study problems in self-organizing systems. A number of uses such as this for general-purpose logical units can be imagined.

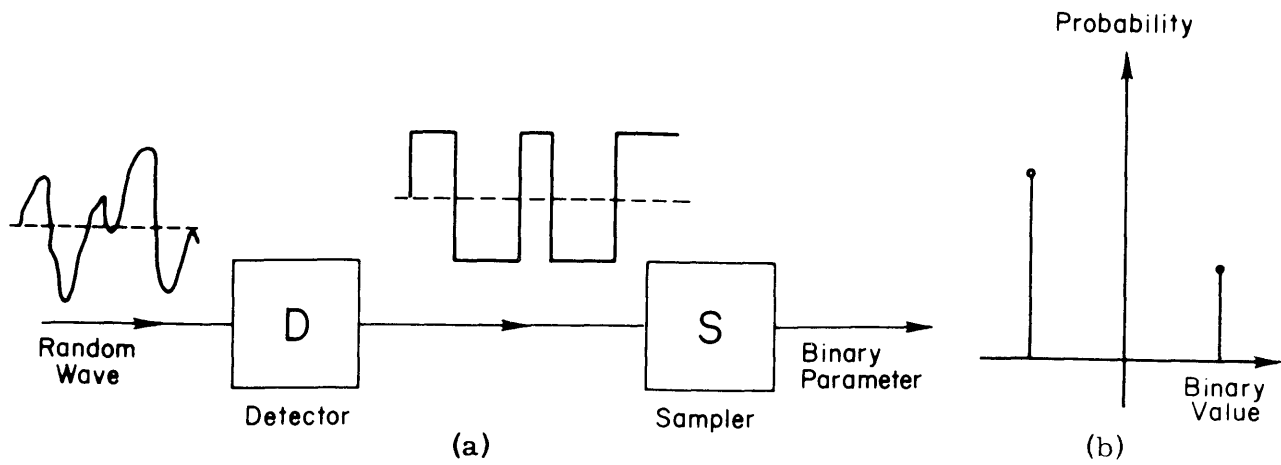


Figure 4-4

Generation of Binary Parameters

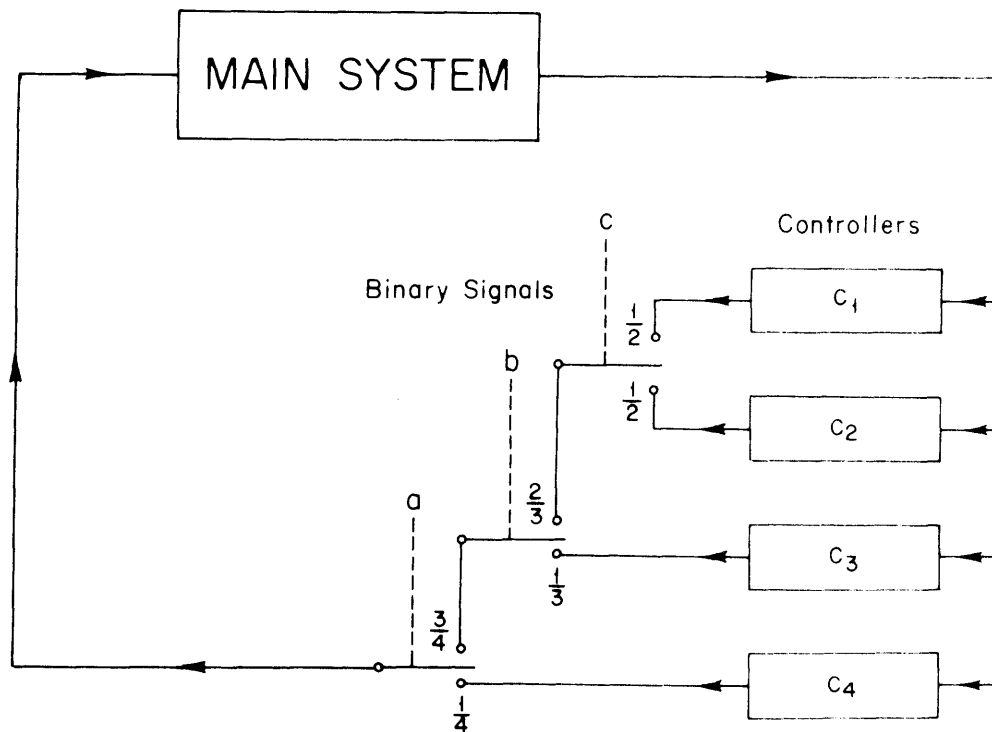


Figure 4-5

Discrete Value Parameters Used for Organization Studies

Memory

There are several types of memory needed for an automatic analog computer and several ways of achieving each category. One type is needed to store a value for a short time. An example is the device which samples a wave during the "set" phase and holds it during "run" and "judge" to represent a trial parameter. In this case a fast follower made out of fed-back integrator that could be switched to a "hold" position would probably be adequate. The drift of this type of memory is unimportant if it is held within moderate limits, because of the short times that values are held in the memory.

In another class of memory devices the drift rate is very important. For example, F^* must be held accurately for possibly long times during which many bad trials may be made. In the time variable case it is important to control the drift accurately.

The goals of fast response when storing a new value and low or precisely controlled drift are antagonistic for an analog memory. It is quite likely that digital storage would be more appropriate for these applications. The number of such long-term memories needed for even rather complex problems is quite small. In many cases it is only necessary to store the $n+1$ values of F^* , x_1^* , x_2^* , \dots , x_n^* .

An entirely different class of memory units are required in Dynamic Extremal Control and might well prove useful for many problems. These are memory devices which can store a function of time at one time scale

and read it back at another time scale. In Dynamic Extremal Control the recent history of inputs, environment, and outputs can be stored in real time, and then played back repeatedly at high speed for model matching and optimization. A mechanical version of such a time-scaling function memory is shown in Figure 4-6.

In Figure 4-6 the data is recorded on a tape at a slow speed, and read off over and over at a much higher speed in such a way that an amount of recorded history of a real time interval ΔT is read off in a span of computer time Δt . The time scaling factor is $\frac{\Delta T}{\Delta t} \approx \frac{v}{V}$. In successive computer epochs the history read off is not quite the same, since the tape has moved at least an amount corresponding to the reading time Δt . Hence if the model matching process is stated as a random scan to minimize a measure of the difference in the responses of the system and model, it will clearly be a time variable problem such as has been discussed previously.

There must be many devices which would perform this type of time scaling -- a slow delay line which could be scanned rapidly would give a sampled version of history. Scaling the other way from computer time seems easier, simply by sampling the repeating computer solutions as in the two time scale paper (Ziebolz).

The use of such time scaling memories will facilitate communication between the machine and real time equipment while retaining the high-speed search capabilities.

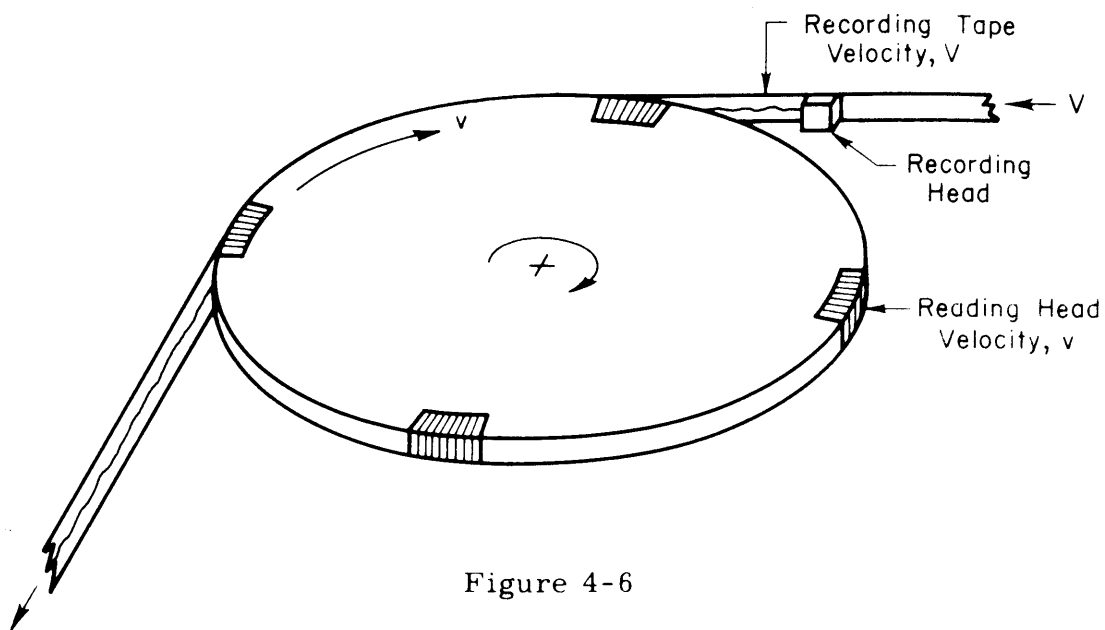


Figure 4-6

Time Scaling Function Memory

Machine Output and Display

In general a high-speed automatic analog computer can be expected to communicate only reduced data to other machines or to human beings in contrast to the usual analog machine which often repeatedly communicates all the information possible. For instance, during a search it is of interest to observe or record the time history of F^* and x^* as well as any histograms or correlagrams which may be used to switch from search to refinement. One may also record something about the trial points. For example, a photographic time exposure of an oscilloscope face on which trial points are plotted can be used for a two-dimensional search or for two-dimensional projections of a higher dimensional search to indicate which regions of the space have been most thoroughly searched.

By means of such simple indicators one abstracts from the mass of trial data most of the information of interest. In any situation many forms of analog processing can be used on the data before it is observed or recorded. An example might be the case in which one wishes to determine the location of a smooth peak with the greatest possible accuracy. One way is to center a narrow search over the approximate location, allow F^* to have a slow drift and to average the x^* position with an analog averager. The effects of noise and random inaccuracy can frequently be eliminated in this way and a very good value of x^* obtained. This technique must not be used in those cases in which x convergence is poor due, for example, to a dithering of x^* between nearly equal relative peaks.

In communication with other machines the analog computer may transmit slowly varying information such as x^* or may time-scale information from the high-speed solutions to real time. An instance of this would be the two-time scale approach of Ziebolz and Paynter in which prediction of the future in model time would be sampled at appropriate times and transmitted to real time equipment. This technique is well discussed in the Ziebolz-Paynter paper and need not be elaborated here. However, in our examples we may wish to use the computer for other purposes than repetitive prediction and to use a time scaling function recorder to record a predicted behavior in model time for use in real time. In this way, for example, the computer could decide on an optimum behavior for the control variables in the near future, record this information, and then go on to other tasks such as model rematching and control reoptimization. Evidently very complicated automatic programming of the sequence of computations to be made can arise in this connection.

The completely automatic optimizing analog computer can be developed progressively. The first step is to remove the details of searching from human interaction. Subsequently, the grand strategies of searching can be done automatically by schemes such as have been suggested. The eventual outcome can be something like the Dynamic Extremal Control in which the machine automatically redesigns a control system in response to changes in the input characteristics, environment, or the system characteristics.

The first of these steps is very simple to achieve. The subsequent steps involve increasing amounts of engineering effort and expense, for not only will special equipment be necessary but the usefulness of the results will be much enhanced if the computer is specifically designed for high-speed operation.

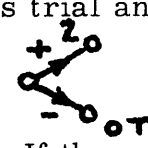
APPENDIX

TWO STRATEGIES FOR LONG DIVISION

The operations involved in decimal division will be discussed here. To slightly simplify the discussion, let us suppose that the quotient has already been calculated to several places, and study the problem of determining what the next digit should be. It will be a number from 0 to 9 depending upon how many times the divisor can be subtracted from the subdividend (how many times 23 "goes into" 103 in the example without having a negative result after subtraction).

$$\begin{array}{r} 2? \\ 23 \overline{) 563.9} \\ \underline{46} \\ 103 \end{array}$$

A mechanical divider would subtract 23 from 103 successively until the result was negative counting the number of operations n , then add back the last 23 and record $n-1$ in place of the "?" and proceed.

This trial and error strategy is diagrammed in Figure A-1. The symbol  means one first subtracts 1 x divisor from the subdividend. If the remainder is positive, one moves along the line and subtracts 2 x divisor. If the remainder is negative the number of times the divisor will "go into" the subdividend is 0, so one goes along the line to OT meaning record 0 and terminate work on this decimal place. If one has gone to 2, 2 x divisor is subtracted, and again there is a branching either to record 1 and terminate or to try 3 x divisor.

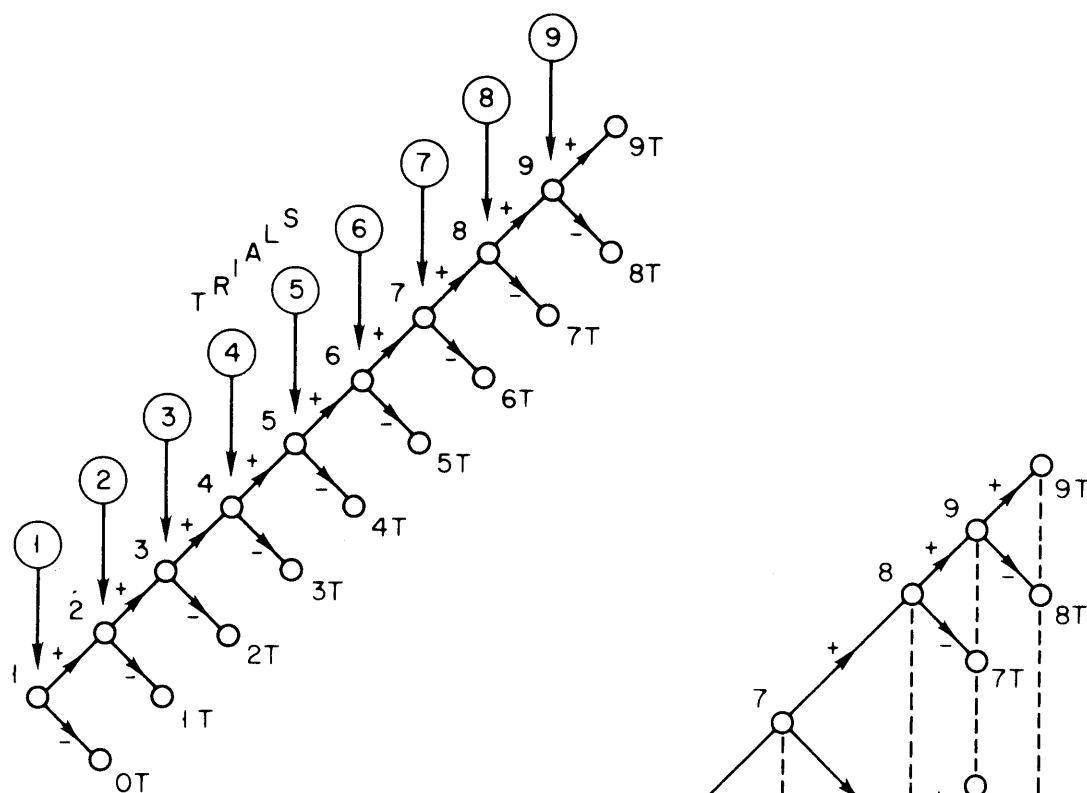


Figure A-1

Standard Division Algorithm

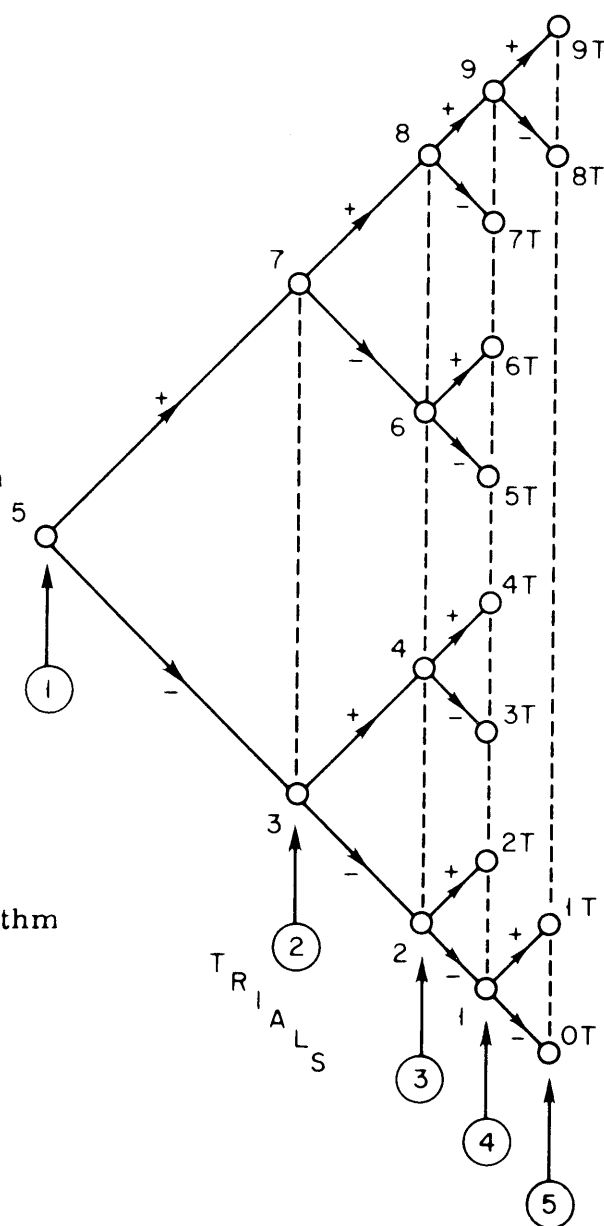


Figure A-2

Minimax Division Algorithm

There is a maximum number of 9 trial subtractions, and if the outcomes OT to 9T are equiprobable, the average number of trials is 5.4.

There is a sort of minimax way to search for the desired number which is very similar to a search for a zero of a monotonic function. Assuming the number to be recorded in the decimal place under consideration, n , can lie at either end point, 0 or 9, it is certainly better to try 5 x divisor first, since this will result in narrowing the search to 5 possibilities no matter what the result of the trial is.

Try 5 then.	0 1 2 3 4		5 6 7 8 9
	Here if		Here if
	remainder		remainder
	is -		is +

Figure A-2 is a diagram of a process based on the attempt to continue cutting the possibilities in half. In this case there will be a maximum number of trials of 5 with an average number of 4.4. This is clearly a better way to do division as long as the probability of terminal results for this digit is nearly the same for all possibilities. A randomized scheme could do even better because at the second trial we have to split the remaining number of possibilities unequally. This results from using 10 as a number base. However, if we used a chance mechanism to choose between 3 and 2, and 6 and 7, so on the average it would be as if we picked $2\frac{1}{2}$ and $6\frac{1}{2}$, the asymmetry in the graph would disappear and the true minimax theory would pertain statistically.

Of course, our minimax division rule assumes that it is no more costly to subtract 5 x divisor than 1 x divisor which is not really true, since the conventional schemes merely add up the divisors to multiply, and our scheme would have to multiply separately. The example is made merely to point out the trial and error and search aspects of a familiar rigid algorithm.

SUMMARY OF EXPERIMENTAL TECHNIQUES AND RESULTS

Many of the random techniques discussed in the main body of the thesis have been accomplished experimentally and will be reported below. The experiments were performed both on a Philbrick Analog Computer and on a piece of demonstration apparatus designed to illustrate some of the problems associated with searching for an absolute maximum or minimum.

A photograph of the demonstration apparatus is shown below in Figure A-3.

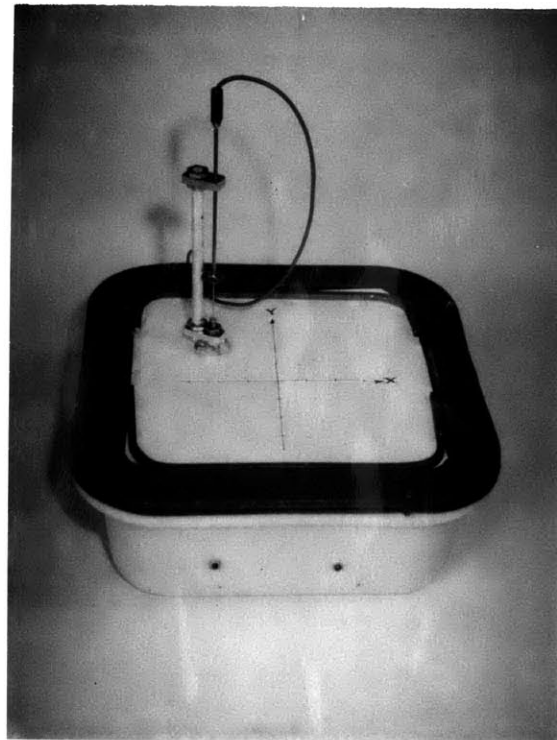


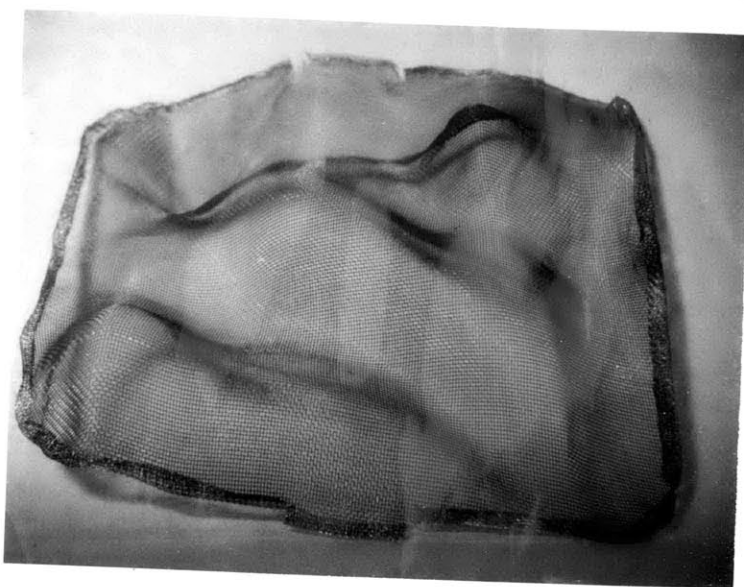
Figure A-3

Demonstration Apparatus

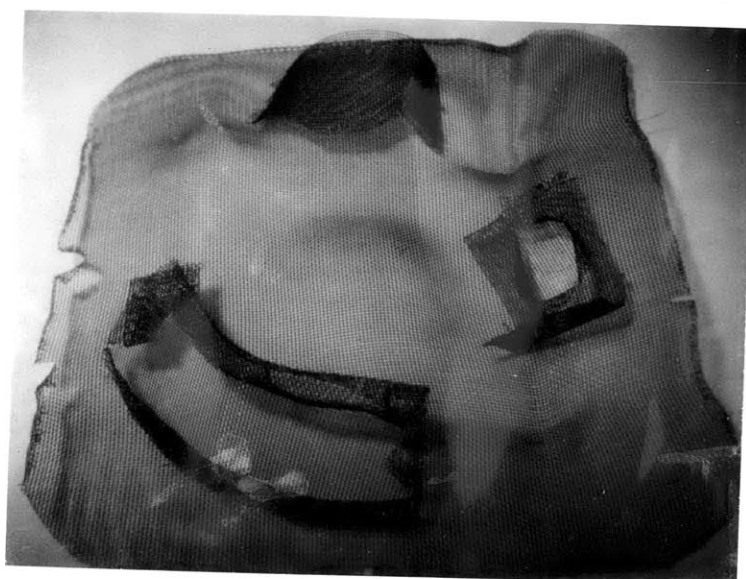
The device consists of a needle which can be pushed down through a piece of thin graph paper suspended on a mesh stretched over the top of a box-like structure. Inside the box, wire screening, bent into hills and valleys to represent a function, is suspended. The distance that the needle can be pushed down into the box before touching the screen is then a function of location in the x-y space shown in Figure A-3. Contact between the needle and the screen function is noted electrically by the sounding of a buzzer. The needle carrier assures that the needle is inserted perpendicularly to the x-y surface and contains provisions for reading F , the distance that the needle has been inserted and for recording mechanically F^* , the greatest distance that the needle has been inserted during a series of trials. Marks can be made on the graph paper to record x^* and the holes in the paper record all the trial x values.

Two of the functions which have been used in the device are shown in Figure A-4. In A-4(a) a smooth, continuous function is shown and in A-4(b) a function which is discontinuous with one narrow, deep spot.

Examples of experimental results obtained with this device are shown in Figure A-5. Figure A-5(a) is a histogram obtained from the function of A-4(a) and A-5(b) was obtained from A-4(b). In each case the histograms were found by using a random number table to generate an equiprobable scan in x-y. The significance of these results is discussed in section 2/2/7. Moving mean refinement techniques of the type discussed in the main body of the thesis were also tried on the apparatus.



(a)



(b)

Figure A-4

Two Screen Functions Used in the Demonstration Apparatus

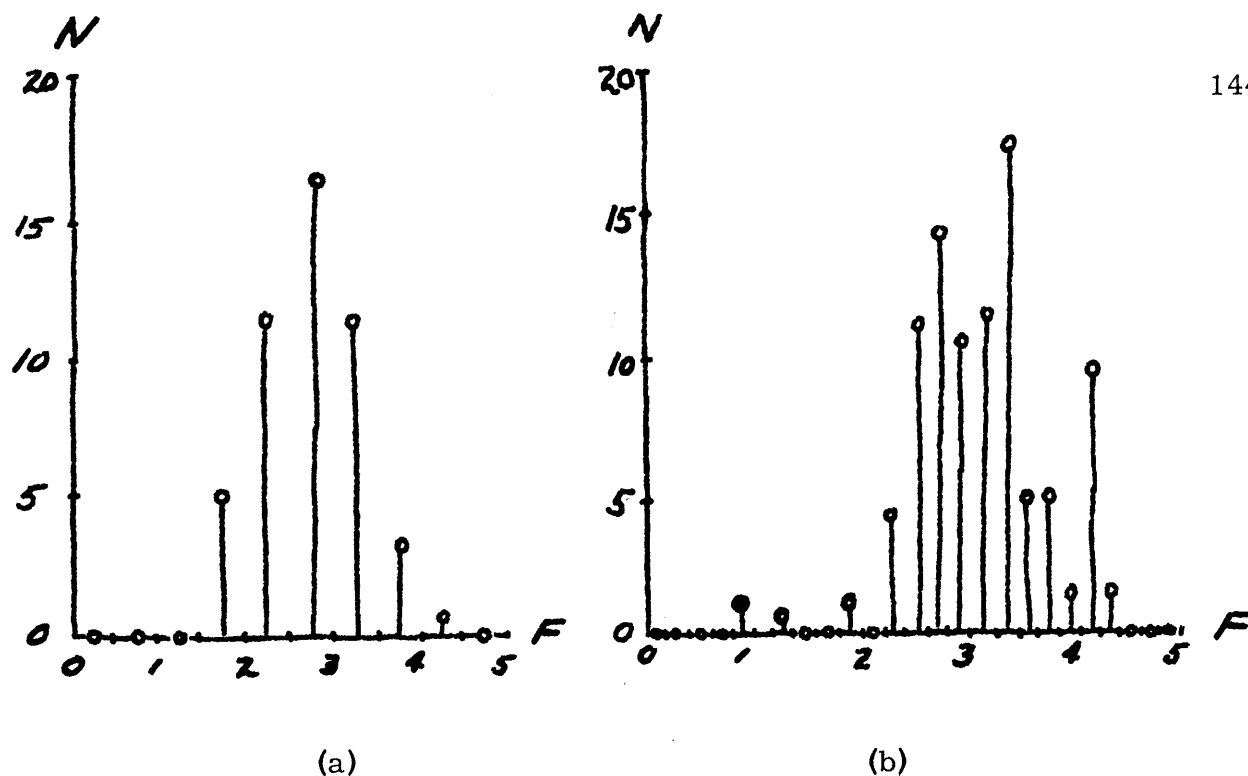


Figure A-5

Experimentally Obtained Histograms

Other experiments were run on the analog computer in the mechanical Engineering Department of M.I.T. and this experience formed the basis for Chapter 4 of the thesis. In particular Figures 4-1, 4-2, and 4-3 are sketches of operations which have actually been accomplished. Several types of static and static-dynamic problems were run. Figure A-6 shows two oscilloscope photographs of a test static function which was used. The function was of the form

$$F(x, y) = \frac{1}{a((x-1)^2 + (y-1)^2) + b} + \frac{1}{c((x+1)^2 + (y+1)^2) + d} + e$$

This function has maxima at about $x, y = \pm(1, 1)$ and the maxima may be equal in value or not depending upon the relative values of the constants a, b, c, d . For the equipment used it was convenient to scale so that one unit in x and y was 25 volts.

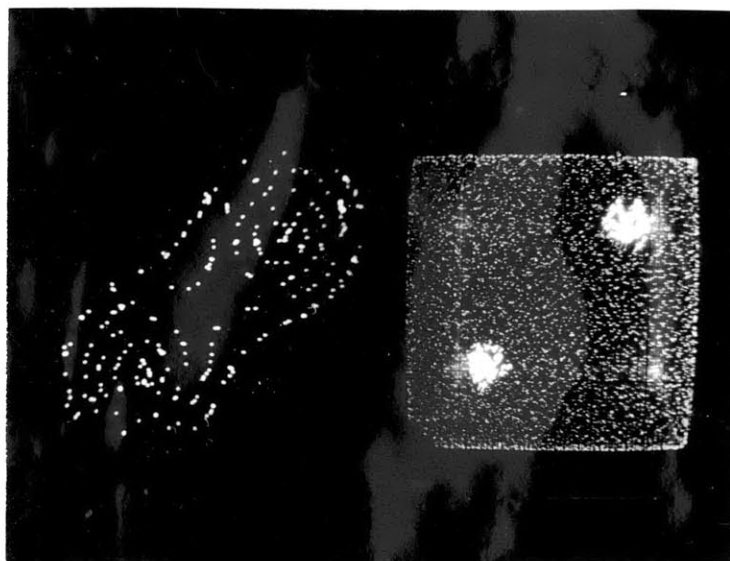
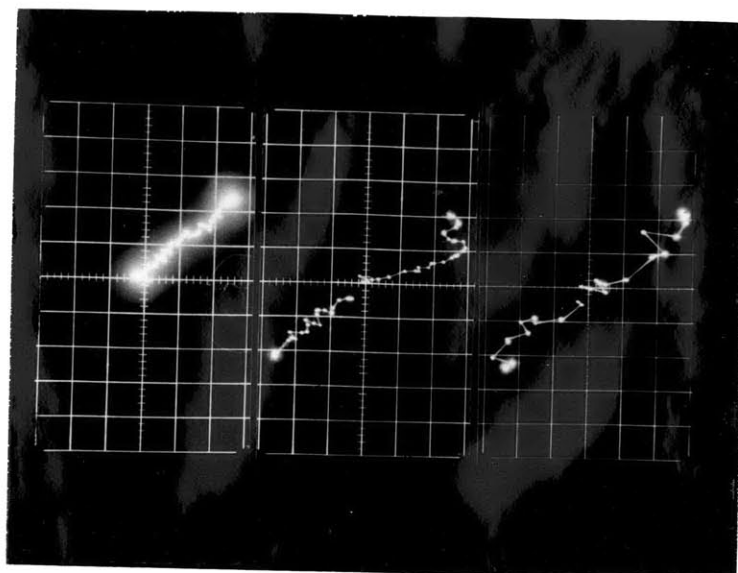


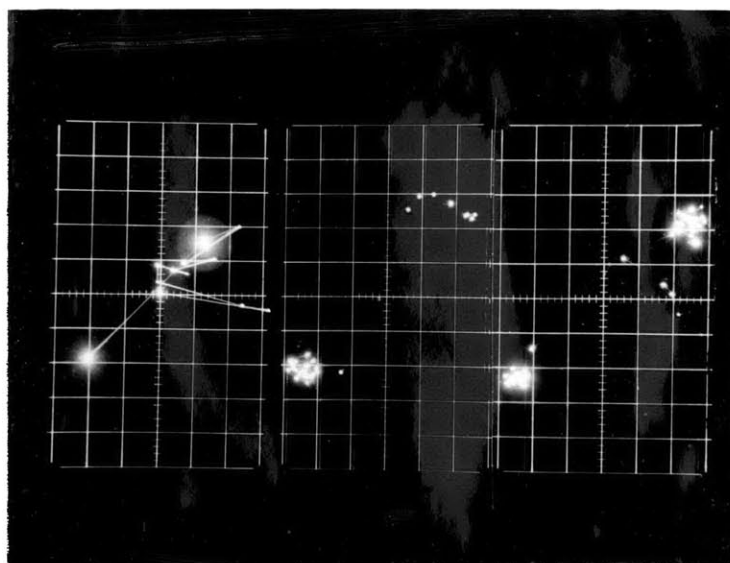
Figure A-6

Static Test Function

The left hand photograph of Figure A-6 shows a contour plot of the function for the equal peak case. This was obtained by plotting x and y on an oscilloscope during a continuous random scanning process and intensifying the spot whenever particular values of the function were computed. The right hand photograph is a record of a random scan in a rectangular x, y space where the spot representing the trial was intensified for high values of the function. In this case the mechanical relays used limited the trial rate to about 100 trials per second. The waves sampled as in Figure 4-2 to generate the trial points were desynchronized triangular waves and the photograph shows that good uniform coverage can be obtained without using strictly random waves.



(a)



(b)

Figure A-7

x* Trajectories from Analog Computer Studies

Figure A-7 shows trajectories of the x^* as the test function is maximized in several different ways. In each of the six photographs, a portion of the x, y space is shown. The large divisions of the oscilloscope represented 10 volts so the maxima of the function occur in the first and third quadrants at about two and a half divisions from the center in both x and y . The three photographs of Figure A-7(a) all show the effect of a narrow moving mean scan. From left to right, the width of the scan in both x and y was ± 2 volts, ± 5 volts, and ± 10 volts. In the last two examples of A-7(a) the process was deliberately started from two biased points so that both peaks were climbed. It is clear that the narrower the scan the more nearly a steepest ascent path is followed. It was also found from observations in time that the wider scans got near the maxima more quickly than narrow ones but once x^* was very near the location of the maximum the narrow scans could refine the solution faster than the wider ones.

The left hand photograph of Figure A-7(b) shows a pure wide scan of ± 50 volts and in this case x^* jumps from peak to peak and after a few initial successes, the refinement of the solution becomes very slow as predicted.

The other two photographs of Figure A-7(b) show the interesting case of the use of a composite scan previously sketched in Figure 2-10 and discussed with reference to Figure 4-3. The peaks of the test function were adjusted to have values of 15 and 13 volts. In both photographs, the composite scan consisted of a moving mean scan ± 10 volts used about 70% of the time and a ± 50 volt wide scan used 30% of the time.

Both photographs show the process initially heading toward the relative peak in the first quadrant but eventually transferring to the absolute peak due to the wide scan. Of course, this sort of a scan often heads for the absolute peak directly.

Another aspect of interest is the dispersion error which is evident in these photographs. This resulted from the necessity of introducing a certain amount of down drift in the storage of F^* in order to make sure that the triggering of the mechanical relays was reliable. Indeed the storage and triggering on the value of F^* was the most critical part of all the analog computer studies.

The static function was also used to study the machinery for static-dynamic problems. One of many ways this was done was to use the trial value of the function as the initial condition of a damped second order system. If the response of the system is then treated as an error signal and one of the usual error criteria is used, the value of the error criterion will be a monotonic function of the trial value of the static function. In this way, the computer solves a static-dynamic problem whose characteristics are well known.

Since much of the computer was used in setting up the functional operations involved in the random scan techniques, no very large real problems could have been done without extensive special equipment.

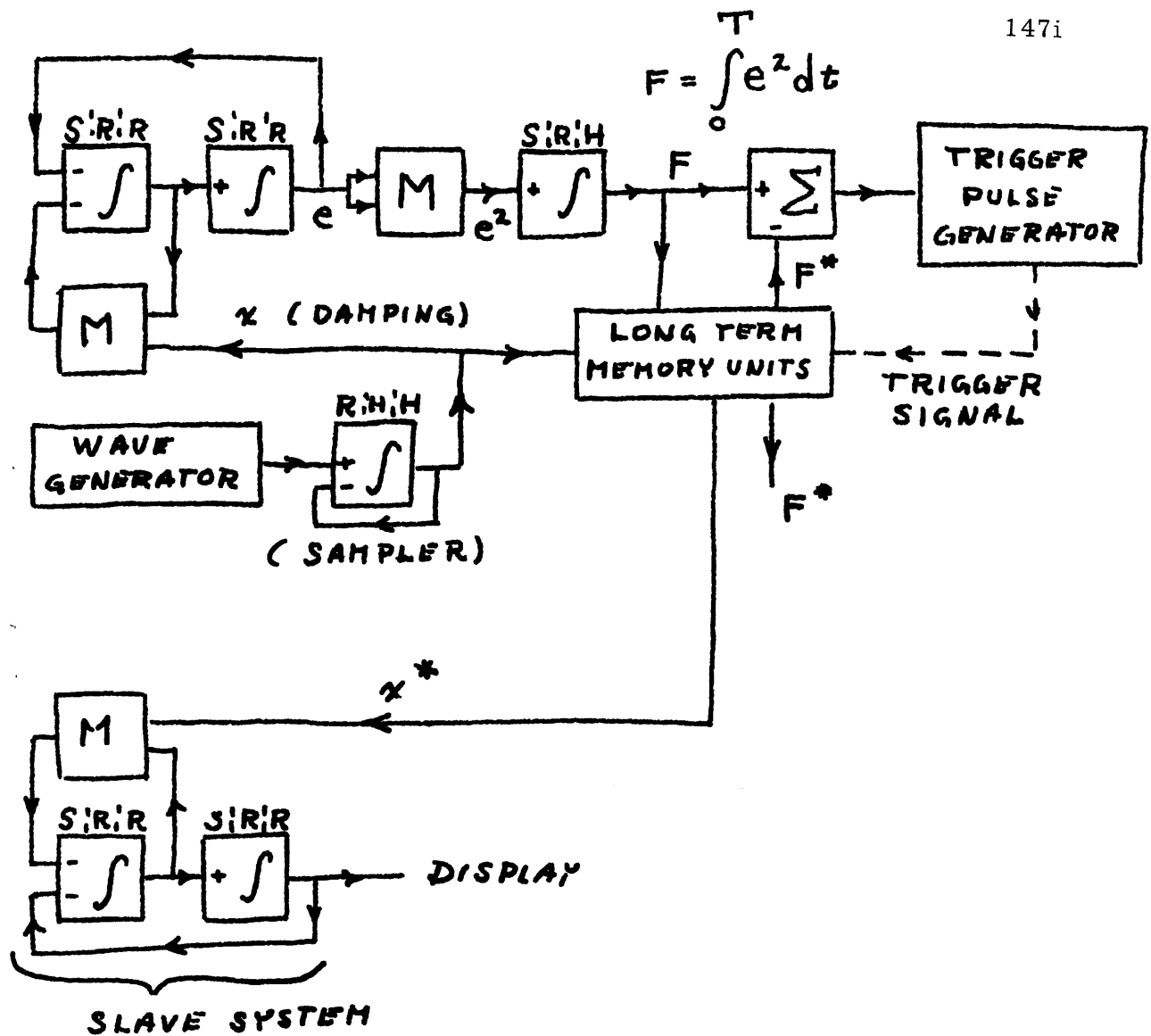


Figure A-8

Symbols Above Integrators

S - set

R - run

H - hold

A One Parameter Static-Dynamic Problem with
Provision for Slave Display

Figure A-8 shows a block diagram for a simple one parameter static-dynamic problem, the problem of finding the optimum damping ratio for a second order system according to the integral squared error criterion. All the integrators have a three phase control cycle as discussed in Chapter 4. In the standard Philbrick computer, it was necessary to use two separate relay controls and clamp signals and some separate relays in order to achieve this effect without altering the control structure of the machine.

From the diagram, one can see how parameters can be set through multipliers and how a sampler can be constructed from a fed-back integrator. The wave to be sampled may be random or periodic if desynchronized from the sampling frequency and may be centered around x^* if the scan is to have a moving mean. To change the scan from wide to narrow on the basis of any criterion, one need only change the characteristics of the wave from the wave generator.

It has been found interesting in problems of this kind to display the output of a slave system whose parameters are set at x^* . By means of this type of display, one can follow the improvement in the system as it is optimized even when the trials are going on at a great rate. Using this type of display, the optimum response of Figure 1-5 has been demonstrated. The reason more work was not done with real problems of this sort was that many of them turn out to have criterion functions which are simple and uninteresting from the point of view of search theory, particularly if only one or two parameters are involved so that the process can be visualized.

BIBLIOGRAPHY

- Abramson, N. M., "Seminar on Decision Theory and its Application to Communication Theory", I.B.M. Research Paper, RJ-MR-8, January 1958.
- Anderson, R. L., "Recent Advances in Finding Best Operating Conditions", American Statistical Association Journal, v. 48, 1953, pp. 789-798.
- Aseltine, J. A., Mancini, A. R., and Sarture, C. W., "A Survey of Adaptive Control Systems", I.R.E. Transactions on Automatic Control, 1958, pp.102-108.
- Ashby, N. R., Design for a Brain, 2d Ed., John Wiley and Sons, New York, 1960.
- Bayes, T., Facsimiles of Two Papers by Bayes, Prepared under the direction of W. E. Deming, Washington, The Graduate School, the Department of Agriculture.
- Bellman¹, R., Dynamic Programming, Princeton University Press, Princeton, 1957.
- Bellman², R., Glicksberg, I., and Gross, O., "On the Bang-Bang Control Problem" Report P-552, The Rand Corporation, Santa Monica, California, 1955.
- Booth, A. D., "An Application of the Method of Steepest Descents to the Solutions of Systems of Non-Linear Simultaneous Equations", Quart. J. Mech. and Appl. Math., v. II, 1949, pp. 460-468.
- Box, G. E. P., and Wilson, K. B., "On the Experimental Attainment of Optimum Conditions," Journal of the Royal Statistical Society, Series B, v. 13, 1951, pp. 1-45.
- Brooks¹, S. H., "A Comparison of Maximum Seeking Methods", Operations Research, v. 7, 1959, pp. 430-457.
- Brooks², S. H., "A Discussion of Random Methods for Surface Maxima", Operations Research Journal, v. 6, 1958, pp. 244-251.
- Bross, I. J., Design for Decision, MacMillan and Co., New York, 1953.

Brown, R. R., "Gradient Methods for the Computer Solution of System Optimization Problems", E. E. Dept. Thesis, M.I.T., 1957.

Bruijn, N. G., de, Asymtotic Methods in Analysis, Interscience, N. Y., 1958.

Campbell, D. T., "Blind Variation and Selective Survival as a General Strategy in Knowledge Processes", Self-Organizing Systems, (M. C. Yovits and S. Cameron, ed.), Pergamon Press, New York, 1960.

Chernoff, H., "Remarks on a Rational Selection of a Decision Function", Cowles Commission Discussion Paper, Statistics, mo. 326A, 1948.

Chichinadze, V. K., "Logical Design Problems of Self-Optimizing and Learning Optimizing Control Systems Based on Random Searching", Proc. of the First Congress of the International Federation of Automatic Control (Moscow, 1960) Butterworths, London, 1960.

Crockett, J. B., and Chernoff, H., "Gradient Methods of Maximization", Pacific Journal of Math, v. 5, 1955, pp. 33-50.

Curry, H. B., "The Method of Steepest Descent for Non-linear Minimization Problems", Quarterly of Applied Math., v. 2, 1944, pp. 258-262.

Draper, C. S., and Li, Y. T., "Principles of Optimalizing Control Systems and an Application to internal Combusion Engine", A.S.M.E. Publications, 1951.

Favreau, R. F., and Franks, R., "Random Optimization by Analog Techniques", Proceedings of the Second International Analog Computation Meeting, Presses Academiques Europeenes, Brussels, 1959.

Flood, M. M., "Sequential Decisioning", Information and Decision Processes, (R. E. Machol, ed.), McGraw-Hill, New York, 1960.

Goodman, L. A., "On Methods of Amalgamation", Decision Processes, R. M. Thrall, C. H. Coombs, and R. L. Davis, eds., John Wiley and Sons, Inc., New York, 1954.

Gregory, P. C. (ed), "Proceedings of the Self Adaptive Flight Control Systems Symposium", W.A.D.C. Report 59-49, Astia Document AD 209389, March, 1959.

Haddad, R. A., and Braun, L., Jr., "Automatic Methods of Process Identification", Adaptive Control Systems, E. Mishkin, and L. Braun, Jr., eds., McGraw-Hill Book Co., Inc., New York, 1961, pp. 293-322.

Hurwicz¹, L., "A Class of Criteria for Decision Making Under Ignorance", Cowles Commission Discussion Paper, Statistics, No. 356.

Hurwicz², L., "A Criterion for Decision Making Under Uncertainty", Cowles Commission Discussion Paper, Statistics, No. 355.

Kac, M., Notes on Lectures Delivered at M.I.T., (unpublished), Nov., 1949.

Kalman, R. E., Ladios, L., and Shapiro, E., "Mathematics is the Key", Chemical Engineering Progress v. 56, No. 2., 1960.

Kiefer, J., "Optimum Sequential Search and Approximation Methods Under Minimum Regularity Assumptions", J. Soc. Indust. App. Math., v. 5, 1957, pp. 105-136.

Leiter, M., "Application of a Digital Computer to the Design of Non-linear Optimum Systems", E. E. Dept. Thesis, M.I.T., 1959.

MacKay, D. M., and Fisher, Analogue Computing at Ultra-High Speed, to be published.

Merriam, C. W., III, "Synthesis of Adaptive Control", Report 7793-R-3, M.I.T. Servomechanisms Lab, 1959.

Meyer, H. A., ed., Symposium on Monte Carlo Methods, John Wiley and Sons, Inc., New York, 1956.

Milne, W. E., Numerical Solution of Differential Equations, John Wiley and Sons, Inc., New York, 1953.

Milnor, J., "Games Against Nature", Decision Processes, (Thrall, R. M., Coombs, C. H., and Davis, R. L., eds.) John Wiley and Sons, New York, 1954.

Minsky, M., "Steps Toward Artificial Intelligence", Proc. I.R.E., Jan. 1961.

Morse, M., "The Critical Points of Functions and the Calculus of Variations in the Large", Bulletin of the American Mathematical Society, v. XXXV, 1929, pp. 38-54.

Pontryagin, L. S., "Some Mathematical Problems Arising in Connection with the Theory of Optimum Automatic Control Systems, (in Russian), Session of the Academy of Sciences of the U. S. S. R. on Scientific Problems of Automating Industry, 15-20 October, 1956, "Basic Problems of Automatic Regulation and Control", Izd. -vo. Akad. Nauk. S. S. S. R., 1957.

Rozanoer, L. I., "L. S. Pontryagin Maximum Principle in the Theory of Optimum Systems, parts I, II, III", Automation and Remote Control, (English Translation), v. 20, 1959.

Sauage, L. J., "The Theory of Statistical Decision", Journal of the American Statistical Association, v. 46, 1951, pp. 55-67.

Selfridge, O. G., "Pattern Recognition and Learning", Proc. Third London Symposium on Information Theory, 1955, Academic Press, New York, 1956, pp. 345-353.

Sommerhoff, G., Analytical Biology, Oxford University Press, London, 1950.

Todhunter, I., "A History of the Mathematical Theory of Probability", MacMillan and Co., Cambridge and London, 1865, pp. v. - 624. (Particularly Articles 690, 719, 806, 990)

Truxal, J. G., and Padalino, J. J., "Decision Theory", Adaptive Control Systems, E. Mishkin and L. Braun, Jr., eds., McGraw-Hill Book Co., Inc., New York, 1961, pp. 479-501.

von Neuman, J., and Morgenstern, O., Theory of Games and Economic Behavior, Princeton University Press, Princeton, 2d ed., 1947.

Wald, A., Statistical Decision Functions, John Wiley and Sons, Inc., New York, 1950.

Wiener¹, N., Cybernetics, John Wiley and Sons, New York, 1948.

Wiener², N., The Extrapolation, Interpolation, and Smoothing of Stationary Time Series with Engineering Applications, John Wiley and Sons, Inc., New York, 1949.

Ziebolz, H., and Paynter, H. M., "Possibilities of a Two-Time Scale Computing System for Control and Simulation of Dynamic Systems", Proc. Nat'l. Electronic Conf., v. 9, 1954.

Фельдбаум, А. А., Вычислительные Устройства в Автоматических Системах, Государственное Издательство Физико-Математической Литературы, Москва, 1959.

Витенберг, И. М., "Вопросы Построения Быстродействующей Электромоделирующей Установки для Анализа Динамических Систем", Вопросы Теории Математических Машин, (под редакцией Ю. Я. Базилевского), Физико-Математической Литературы, Москва, 1958.

BIOGRAPHICAL NOTE

The author was born in Milwaukee, Wisconsin, on June 12, 1934. He attended the Wauwatosa (Wisconsin) High School and graduated in June, 1952. He subsequently entered the Massachusetts Institute of Technology and received the degrees Bachelor of Science and Master of Science in Mechanical Engineering simultaneously in June of 1957.

Since September of 1957 he has been on the teaching staff of the Mechanical Engineering Department of the Massachusetts Institute of Technology and at present he is an Instructor.